1991-09

Design and implementation of Control Command Check System (CCCS) a multimedia DBMS for security applications

Leontakianakos, Ioannis M.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/28101

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DESIGN AND IMPLEMENTATION OF
CONTROL COMMAND CHECK SYSTEM (CCCS)
A MULTIMEDIA DBMS FOR SECURITY APPLICATIONS

by

Ioannis M. Leontakianakos

September, 1991

Advisor: Dr. Thomas Wu

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Computer Science Dept. Naval Postgraduate School; 6b. OFFICE SYMBOL *(if applicable)* CS	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000	7b. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION; 8b. OFFICE SYMBOL *(if applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*

DESIGN AND IMPLEMENTATION OF CONTROL COMMAND CHECK SYSTEM (CCCS) A MULTIMEDIA DBMS FOR SECURITY APPLICATIONS

12. PERSONAL AUTHOR(S)
Ioannis M. Leontakianakos

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
Master's Thesis	FROM 09/90 TO 09/91	September 1991	99

16. SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the United States Government.

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	Multimedia DBMS, Voice recognition, Bar code, Software Tools, Windows Environment, Microcomputers.

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

In many applications, in addition to formatted data which can be managed by the current traditional database management systems, media data such as image and sound are also needed. One such application is a database system for maintaining person's entrance and exit from the building for security reasons. Best solution for such application requires the use of a multimedia DBMS to manipulate data such as formatted data, image, graphics, bar codes and voice input. Although most organizations today own general purpose microcomputers (IBM, compatibles), the implementation of a true multimedia DBMS that can run on microcomputers is not feasible. This creates a problem of not utilizing the existing microcomputers and software. In this thesis, we will design and implement a system for handling multimedia data by using software tools for the microcomputers. Our prototype system runs on general purpose microcomputers.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. C. Thomas Wu	22b. TELEPHONE *(Include Area Code)* (408) 646-####; 22c. OFFICE SYMBOL CS/Wq

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Approved for public release; distribution is unlimited.

Design and Implementation of
Control Command Check System (CCCS)
a Multimedia DBMS for Security Applications

by

Ioannis M. Leontakianakos
Lieutenant, Hellenic Navy
B.S., Naval Academy of Hellas, 1981

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

ABSTRACT

In many applications, in addition to formatted data which can be managed by the current traditional database management systems, media data such as image and sound are also needed. One such application is a database system for maintaining person's entrance and exit from the buildings for security reasons. Best solution for such application requires the use of a multimedia DBMS to manipulate data such as formatted data, image, graphics, bar codes, and voice input. Although most organizations today own general purpose microcomputers (IBM, compatibles), the implementation of a true multimedia DBMS that can run on microcomputers is not feasible. This creates a problem of not utilizing the existing microcomputers and software. In this thesis, we will design and implement a system for handling multimedia data by using software tools for the microcomputers. Our prototype system runs on general purpose microcomputers.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

A. BACKGROUND

The rapid installation of powerful microcomputers increases productivity. Increasing the productivity of people who use these machines is important too. Analysts have estimated that, in over 95 percent of human-computer interactions, labor costs are greater than the machine costs [INFOTECH79]. Thus, user-friendliness of an interface is the key issue.

Research in voice recognition (VR) systems has been ongoing for over 30 years. Already many applications use voice input to computer systems. The development of the bar code industry from the perspective of industrial application, which began in the early 1970s, moved from the laboratory into a significant part of American and International business and industry, promises more innovative and broadly-based applications in the future. In addition, the development of Database Management Systems manages databases and enables end-users to interact with the data. Finally, the development of different types of scanners enables users to enter any document into databases with minimum effort. Although most organizations own powerful microcomputers, their personnel still do a lot of work manually.

B. THE EXISTING PROBLEMS

In every organization there are some areas where entrance is only allowed to authorized persons; such as Communication Centers, Ammunition Stores in military bases, Customer Service Stores and Laboratories. For these areas it is necessary to check the entrance/exit of every person, verify his/her ID, check the time of entry/exit, note the duration of all visits, and finally store all information. In the past, this process has been done by *security guards*, who simply check the visitor's ID card. However, to accomplish a good security level, an authorized person is provided with an ID that contains a lot of confidential information (Photo, Rank, Person's record, Person's duty in the area, etc). In such a way, *security guards* are able to have the overall picture of the visitor and check information easily. Unfortunately, the loss of such an ID might provide an unauthorized person with confidential information, which would create security problems.

In addition, in many organizations there are departments/equipments that must be checked at regular time intervals. Areas such as ammunition stores, engine rooms of a ship, hold of a ship, research laboratories, and nuclear factories are checked by specific *safety guards*, for safety purposes. Until now, this check has been done manually by *safety guards* at on a fixed period of time; using physical keys. The *safety guard* keeps a clock-counter and visits each department at regular intervals of time. In every department, using the physical key, he stores the department ID and time of check on a paper strip inside the clock-counter. The next day the control officer takes out the paper strip, checks all the key inputs to see if the *safety guards* have done their job correctly, and stores the information.

Is there a way, to improve these manual check security/safety systems, using the existing microcomputers?

C. A POSSIBLE SOLUTION

A possible solution to the previous problem is the use of a multimedia Database Management System (MDBMS) which would enable end-users to interact with all different kinds of data concurrently. A multimedia DBMS which can manipulate:

- formatted data (i.e., all the data types supported by a traditional DBMS)
- images (ex., person's photo, equipments' photo)
- graphics (ex., departments' drawing)
- voice recognition systems
- bar codes

By definition a multimedia Database Management System is the DBMS [LUM88], [LUM89] that can manipulate data such as:

- formatted data
- images
- graphics
- sound (ex., a sample of person's voice)

The first approach of MDBMS definition differs slightly from the second approach. The first approach does not contain the item **sound**, but it contains two new items; **voice**

recognition system and **bar codes**. However, that isn't a significant problem, since a multimedia DBMS for a specific application does not need to manipulate all of the previously referred items necessarily. Moreover, the bar code input can be viewed from two different perspectives. One as a means to enter data into computers (i.e., input data device); two the entered data defines some value of a records' attribute, which uniquely identifies the associated record (ex., person's ID). Thus, bar codes can be considered to be a part of formatted data for the needs of DBMS. The voice recognition system is also an input device. But it differs significantly from the traditional input devices, since the input is the user's voice which, after it is interpreted, is converted to specific commands that the computer has to execute. Since these commands are related to the DBMS, a MDBMS has to consider this kind of input too.

D. A BETTER SOLUTION

Of course, the previous approach is possible, but it seems to have disadvantages. First we need to have a multimedia DBMS which would manage the previously referred data, that is not presently available. Moreover, the design and implementation of such multimedia DBMS might need a specific system on which it is run. That creates the problem of not using the existing microcomputers and significantly raises the overall cost.

To solve the problem described in the previous paragraph, by exploiting the existing microcomputers at organizations, we can develop a prototype application combining dBASE III PLUS, ToolBook, MS Windows, a Voice Recognition Input System, a Scanner, and a Bar Coding System. This prototype, **Control Command Check System**

(CCCS) provides a better security/safety level, and is easy to use due to the well designed graphical user interface.

E. OUTLINE

This research will present one major topic per chapter. The major objectives of this thesis are:

- how we implement a multimedia DBMS that manipulates data of traditional databases, scanned photographs, voice inputs and bar coding inputs;
- how such a system could be implemented on the IBM compatible personal computer, using existing commercial software;
- how this system could satisfy both novice and experts users; and
- how such a system improves the security/safety level;

These issues will be discussed together with a detailed description of the prototype developed for this research. The thesis is organized as follows:

Chapter II contains the related literature. It introduces bar codes, voice recognition systems and the software environment that will be used for the application. First, besides the basic description of bar coding, it also provides a brief comparison to other types of machine-readable labels. Second, it introduces speech, categories of speech recognizer and speech applications. Finally, it discusses the main software packages that will be used to create the prototype software environment, such as MS Windows 3.0, ToolBook and dBASE III PLUS.

Chapter III describes how the designed prototype, **Control Command Check System**, accomplishes the previous objectives. Specific constraints of this application and whom it is designed for are also discussed.

Chapter IV discusses the prototype application **Control Command Check System**. It provides detailed descriptions of the user's environment, presents all the types of screens, and explains how users can navigate through the CCCS and exploit all the provided capabilities.

Chapter V contains the results of this research, examines if the goals, we originally defined, are accomplished, and discusses the performance the CCCS achieves. Issues such as memory space, CPU speed, and environment requirements are discussed too.

Finally, Chapter VI addresses the areas of this thesis for improvement and the areas for future research.

II. LITERATURE

A. BAR CODES

A bar code is a printable machine language. In fact, bar codes are the only printable machine language which reproduces directly the bit-stream of ones and zeros which are the basis for the internal logic of all digital computers. They are messages where information is encoded using the widths of printed bars, the widths of spaces between bars and the relative positions of wide or narrow bars and spaces (i.e., unique wide or narrow combinations of black and white bars). They provide a means of creating labels which can be read by instruments.

Bar coding is a memory form. Printing black bars on white paper is directly analogous to recording plus or minus bars in a magnetic medium. In fact, the basic formats used in these two technologies are identical. While information recorded in magnetic media can be packed at higher densities, and can be erased and recorded, printing bar codes on plain paper is much less expensive for many memory applications.

Bar coding is a communication technique. Just as printed documents function as a basic means of communication between human minds, so bar-coded documents can link widely distributed instruments of diverse kinds into inexpensive communication networks.

Bar coding involves four independent but interrelated processes: printing; reading; transforming from the space phase (what is printed) to a time phase, followed by an algorithm to decode the time phase; and a coding scheme which facilitates all the other

processes. Existing commercial devices introduce major variables into each of these processes.

In one common practice, humans examine each passing item, then communicate their observation to computers via manually operated keyboards; skilled typists can enter messages at a rate of four to six characters per second. As an alternative, where each item is made identifiable by a machine-readable label, that label is read automatically at each system node, key entry is bypassed, and computers receive accurate information in less time and at lower cost.

With bar-code label, relatively unskilled individuals using hand held bar-code wands can achieve a data entry rate of at least twenty characters per second with one fluid movement of the hand. But perhaps of more importance than speed is accuracy. Therefore an automated item-identification scheme is the basis for maximizing the effectiveness of database management systems.

There are three basic types of machine-readable labels now under consideration for discrete-manufacture applications [BURKE84]: optical character recognition (OCR), magnetic strip and bar code. Optical character recognition is the ability of an instrument to recognize or read human-readable characters.

Of the three types listed, magnetic strip and OCR are definitely restricted in their capabilities; only bar coding can be used as a generally applied labeling technique. According to the published literature, OCR substitution error is 1/10,000 for restricted character set. This in itself is so bad that a full character set is seldom attempted.

Generally, the advantages and disadvantages for each label technique are presented in figure 2.1 [BURKE84].

BAR CODE READING	OPTICAL CHARACTER RECOGNITION	MAGNETIC STRIPE READING
	ADVANTAGES	
- EASY TO PRINT - INEXPENSIVE MEDIA - EASY TO COPY - NON-RESTRICTED FORMAT - WORD PROCESSING COMPATIBLE - INEXPENSIVE TO READ - LOW ERROR RATE - HIGH SPEED PRINTABLE - NON-CRITICAL WANDING - MATERIAL IMPRINTABLE - FULL CHARACTER SET - BEAM SCANNABLE	- HUMAN READABLE - EASY TO PRINT - EASY TO COPY - WORD PROCESSING COMPATIBLE - HIGH INFORMATION DENSITY	- READ-WRITE CAPABILITY - LOW ERROR RATE - NON-CRITICAL WANDING - FULL CHARACTER SET
	DISADVANTAGES	
- LOW INFORMATION DENSITY	- EXPENSIVE TO READ - ORIENTATION CRITICAL - RESTRICTED WANDING SPEED - RESTRICTED CHARACTER SET - HIGH ERROR RATE - LOW THROUGHPUT - NOT BEAM SCANNABLE	- EXPENSIVE MEDIA - NOT HUMAN READABLE - MODIFIABLE - WORD PROCESSING INCOMPATIBLE - DIFFICULT TO COPY - RESTRICTED FORMAT - LOW PRINT RATE - NOT READ THROUGH PLASTIC COVER - NOT BEAM SCANNABLE

Figure 2.1 Advantages/Disadvantages of Label Techniques

Security, machine-readability, and various levels of user sophistication (and lack thereof) coalesce in a requirement for an inexpensive technique which can be used to prepare documents carrying messages. Application include amusement park passes, garment tags, electronic hotel keys, transportation vouchers, ID cards and the like [COE84], [LOEFFIER84]. In most applications, bar coding offers a much simpler answer with more read reliability at lower document cost.

B. VOICE RECOGNITION SYSTEMS

1. Speech Composition

Human speech is a complex, well-defined process of conveying information. The process starts with the brain, which sends signals to those muscles and organs used to make speech. The formation of speech sounds then occurs and the process ends with interpretation by the listener. This section will provide a basic foundation for understanding the way speech is formed, the composition of speech signal, and the information components of speech.

The physical process of communicating is achieved by the interaction of lips, tongue, and teeth. Five types of speech sounds articulated in English are [Jensen88]:

- **Plosives** which are sounds created by stopping the passage of air. An example is the letter "t" in the word "top".
- **Fricatives** are caused by forming a narrow passage through which air may pass. The diphthong "th" in the word "their" is an example.
- **Laterals** are sounds formed when the tongue touches the roof of the mouth. An example is the "l" in "launch".
- **Trills** are caused by the rapid vibration of one of the articulators (lips, tongue, etc). The letter "r" is a trill sound in some languages.
- **Vowels** are those sounds made when unobstructed air passes over the vocal cords.

Human speech, then, consists of strings of phonemes, which are the atomic units of sound. Most spoken languages require between twenty and sixty phonemes [Lea80]. Analysis of the phonemes required for a word viewed in isolation is not sufficient because words sounds change depending upon the location within a string of

words. A language's phonological rules govern the phonemes associated with a specific word depending upon other sounds immediately preceding and following the word.

Speech understanding is not based on word alone. Understanding requires not only knowledge about what was said but also how it was said. Hearing phonemes is the basis for what was said. Interpreting the stress, tempo, placement, and duration of pause and intonation implies how it was spoken. This process is termed *prosodics*. An example would be understanding the implication of the following sentences:

"I can see a head." vs "I can see ahead."

The sentences contain identical sounds, yet the prosodics of speech avoids the obvious ambiguity caused if pauses were not considered in the interpretation of what was said. Frequently though, prosodics alone is insufficient for understanding, as in the case of poor enunciation. Resolution of ambiguity may also involve an understanding of context in which a phrase was spoken, which is the term *pragmatics*.

Human speech is also governed by a structure we know as grammar. The grammatical structure is represented by syntax. English syntax, for example, requires a proper sentence to be composed of a noun and a verb phrase. The syntactic rules, in conjunction with prosodics, govern how an utterance may correctly spoken. Linguistictheory suggests the more complex the syntactic constructs, the more powerful the language.

The human process, then, of semantic analysis of speech is reliant upon not only hearing the strings of phonemes but also using the prosodics, pragmatics, and syntax

of the language in order to understand not only what was said but what was meant. This ability allows us to uniquely process phrases such as "up in arms" and "over the hill".

Depending upon the application, speech systems may offer varying degrees of sophistication-from the simple phoneme interpreter (an isolated word recognizer) to a system capable of resolving prosodic and semantic ambiguity (a natural language processor).

2. Speech Analysis

Understanding how speech is analyzed by a machine is simplified by developing parallels between the more familiar human process and the unfamiliar machine process. A *Knowledge Source* is the relative maturity of the system, human or machine. Just as children can be "programmed" to understand, so can a machine. The sophistication or robustness of speech analyzer then is directly related to its ability to process the variety of speech information (phonological rules, prosodics, syntax, and pragmatics).

A fundamental algorithm for understanding what was said is found in figure 2.1 [REDDY76]. This is a classic speech signal analysis algorithm that most processors use, regardless of the technology involved. Conversion of the human analog signal to a discrete digital signal in a machine-acceptable format is the first converter, an attempt is made to bound the signal. Accurate detection of the boundaries of a signal is essential if recognition is to be achieved. Because the entire spectrum of the signal may not be required, an algorithm is employed to isolate the essential signal characteristics. The remainder of the signal is discarded in a process known as data compression. The probability that two utterances of a word or phrase are identical is remote. All recognizer,

then, must be capable of eliminating slight variances in speech, pitch, intonation, and pause length. The filtering or "normalizing" process allows for a range of signal variability. The more robust the recognizer, the greater the variance. Depending upon the mode (learning or recognition), an attempt is made to either add the signal to a vocabulary or match the sound against an existing vocabulary.

Algorithms used to match the signal have been a major research area, while increasing both speed and accuracy is a primary goal. Generally, though, matching is achieved by comparing distances between the incoming pattern and some previously stored reference pattern. The pattern with the minimum distance is judged the winner.

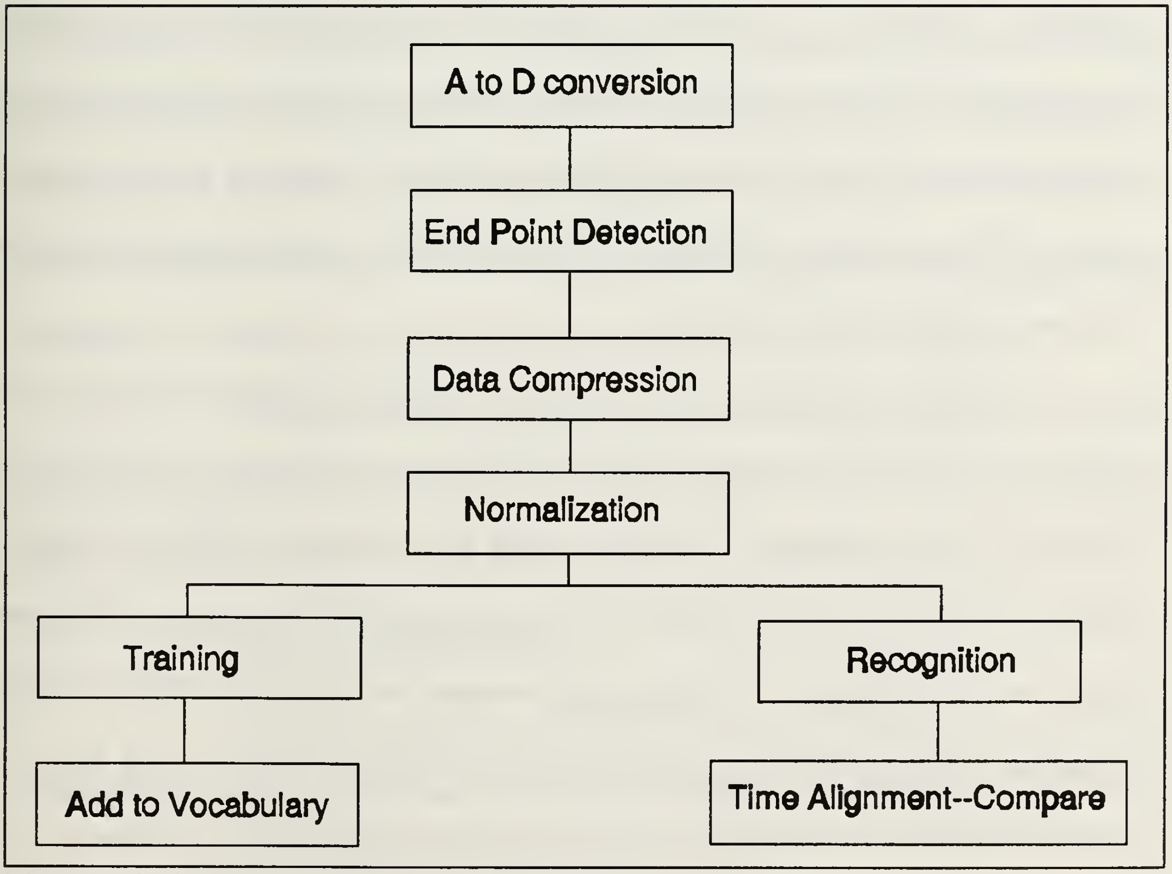

Figure 2.2 Speech Recognition Components

3. Categories Of Recognizer

Research and commercial endeavors have combined to develop a variety of recognizer, which are designed to satisfy specific application requirements. Two points to understand when evaluating any speech recognition system are the degree of speaker independence and how utterances are parsed.

Speech systems today are either speaker independent or speaker dependent. The more common, speaker dependent systems require the user to pre-train the system prior to use. Training typically involves creating a personal template for each word in the vocabulary. Creating a personal speech template for each word in the vocabulary ensures consistent input will be acceptable regardless of individual speaker characteristics. Unfortunately, for connected speech systems with large vocabularies, this could become a time-consuming process. Speaker independent systems employ a standard template against which all speech is compared. The cost is generally a more restricted vocabulary and lower overall recognition rates.

Utterance parsing governs how the recognizer algorithm will dissect the utterance. In isolated systems the recognizer has no syntactic knowledge source, thus each utterance is viewed singularly. Examples would be commands "ENTER" or "DIAL". Short macro phrases are also possible in isolated systems. For example, a recognizer could train to recognize and subsequently execute the command "DIAL HOME". Connected systems, however, view the speech in terms of a syntax, thus strings of commands/words may be spoken in a connected pseudo-language that is a subset of a true

language (e.g. English) for a particular environment. An example might be the command "DIAL PULSE FOUR ZERO EIGHT FIVE FIVE FIVE ONE THREE TWO LOG IN GUEST". An extended variant of connected systems are those that recognize in a continuous fashion, typically according to some natural language syntax. Speech-to-text applications typically employ a continuous recognizer. As a general rule, the more powerful and complete the syntax is, the more natural the interface will be.

4. Speech Applications

a. General

A parallel may be drawn between the development of telegraph/telephone systems and computers in general. The discovereeg by A.G.Bell that speech could also be transmitted via wire caused the replacement of a keyboard with a voice-actuated receiver/transmitter as the primary means of transmitting short duration messages. Why did that occur? The primary reason is that despite our sophistication, voice remains our most natural communication medium. Keyboards are an outgrowth of the typewriter and telegraph technologies but they, too, are limited by the skills of the operator.

Numerous studies have shown that voice recognition systems are faster and more accurate than most manual-entry systems. Additionally, voice systems free the operator's eyes and hands to accomplish concurrent tasks [WOODARD83].

b. Commercial Applications

With the increasing sophistication and decreasing cost, commercial applications of speech systems have surfaced. The variety of applications is only limited

by the imagination. But in the commercial environment, voice input is generally being used for one primary purpose: to increase individual productivity. A typical commercial speech application is in the area of quality control and inspection. Other commercial applications that have been successfully installed include: voice applications of process control, warehousing functions, automated material handling, and parts programming for machine tools [MARTIN76].

c. Military Applications

The employment of speech in a number of mission-critical military systems has increased dramatically in the last decade. Speech recognition research and development has been largely supported by military organizations. Military speech recognition research efforts have been focused into four primary areas: security, command and control (C^2), data transmission and communication, and processing distorted speech [WOODARD84], [DRAKE88].

C. SOFTWARE ENVIRONMENT

One major advance in PC technology is the graphical user interface (GUI), which is becoming the standard for today is PC interfaces. The first (in the early 1980s) graphical interfaces made computers easier to use because their metaphors and icons offered more powerful, flexible, and direct representations of tasks and data than previous, character-based systems. They also required high performance processors and high quality display hardware. Today, Microsoft Windows, OS/2 Presentation Manager, the Apple Macintoch, and other GUI systems deliver similar abilities to the PC users.

Another major advance in software technology is the development of object-oriented programming. Finally Hypertext and Hypermedia represent another new idea for computer users. A hypertext author has a strong set of tools for linking related pieces of text and graphics with video and sound. By traversing the links an author has created, a reader can browse through related information and see conceptual illustrations alongside text.

An operating environment for IBM and Compatible PCs that has all the previously referred features is MS Windows 3.0. In addition many powerful, based on or supporting Windows, programs enable users to use PCs more productively.

1. MS Windows 3.0

MS Windows 3.0 [WINDOWS90] is a multitasking graphical operating environment that makes the use of a PC more productive and more enjoyable. It simplifies the operation of every program, makes complex tasks involving several programs to seem very simple, and improves the printed output of all user's work. It also allows software developers to build better applications than would develop under character-mode DOS. Furthermore, it offers multitasking, a graphical interface, and more powerful and flexible applications than the character-mode DOS word does. Finally the Dynamic Data Exchange (DDE) feature of Windows permits programs operating under Windows to exchange diverse types of data. Any program based on Windows that supports DDE can share its information (be in text, data, or graphics) with any other DDE-aware program.

2. dBASE III PLUS

It is a remarkably straightforward database system [dBASE86], that provides strong support for the user who wants to develop relational database application (although, it is not a true relational system). It became the standard for relational databases. Its file format is recognized by most of the software application tools.

3. ToolBook

ToolBook [TOOLBOOK90], is an application tool that in cooperation with Windows GUI, creates a powerful new metaphor for the PC user. Having the capabilities of drawing, database functionality, text manipulation, application programming and other features, ToolBook provides the user with flexibility for building solutions to unique application needs. By combining easy-to-use programming tools in ToolBook with Windows' graphical text display and ability to control video and sound devices, users are able to create hypertext and hypermedia documents.

III. THE DEVELOPED MULTIMEDIA PROTOTYPE SYSTEM

A. PREFACE

1. The Prototype

The **CONTROL COMMAND CHECK SYSTEM (CCCS)** is a software prototype package. It has the capability of combining the use of speech recognition, bar code scanner, and a database to assist in keeping track of individuals who have access to a secure building.

A command or organization that has to control access to a building can use this system to verify that an individual can be granted access. The command or organization is responsible for the access records.

2. End Users

The system was designed with both novice and expert users in mind [SMITH86] and [SNEIDERMAN87]. It was designed to be used by personnel in entry level positions. This includes very junior enlisted personnel, and security clerks/guards with basic clerical skills.

When an individual wants access to the building, he/she presents a card with a unique bar code on it (person's ID number) to the guard. The guard will, then, either use a scanner that will read the bar code on the card, or enter the individual's ID number using the keyboard. The name, rank, social security number (SSN), and any other information about that individual will appear on the computer screen. The user has the

option of using voice recognition and bar code scanner to input data, or replacing any of the two previous means with the combination of keyboard and mouse.

3. A Graphical User Interface

Studies have shown [SNEIDERMAN87] that graphical user interface makes a substantial difference in learning time, performance speed, error rates and user's satisfaction. Of many different approaches tried, the graphical interface has the widest acceptance. The graphical interface allows the user to make selections with one keystroke or one click on a mouse. In our case, the largest number of users are persons with minimal clerical skills, and the rest of them occasional users. The graphical interface, therefore, guarantees better results than an interface that requires typing in often cryptic and difficult to remember commands.

4. The Eight Golden Rules Of Dialogue Design

The system has been designed and implemented following the eight golden rules for a good user interface [BROWN89]:

- Strive for consistency;
- Enable frequent users to use shortcuts;
- Offer informative feedback;
- Design Dialogues to yield closure;
- Offer simple error handling;
- Permit easy reversal of actions;
- Support internal locus of control; and

- Reduce short-term memory load.

B. DESIGNING THE CCCS PROTOTYPE

1. Three Levels Of System Access

The CCCS prototype has been designed with three levels of access. The high level access is associated with the letter U (real ***Users***), the intermediate level access is associated with the letter R (***Readers***), and the low level access is associated with the letter G (*Guards*). Users in the first category (U) are allowed to enter information into the system about individual's entries/exits, access all the provided information by the system (person's records, user's records, reports, etc), and maintain the existing databases by adding, deleting, updating data. Users in the second category (R) are allowed to enter information into the system about individual's entries/exits, and access some of the information provided by the system, in order to check if the security clerks (G) have done their job correctly. Finally, *guards* (G) can only enter information into the system about entries/exits.

The CONTROL COMMAND CHECK SYSTEM offers to users the capability of using the system into different ways. First, users of all levels can access and manipulate the system using a keyboard, a mouse, a bar code scanner, and speech recognition (whether or not the two last input devices are available). Second, the *guards* (G) can also access the system and enter information about entries/exits using only voice recognition and bar code scanner.

2. Speech Recognition And Bar Code Combination

The capability of entering data for entries/exits into the system using voice recognition input combined with the input of the bar code scanner id offered only to the *guards* (G), figure 3.1. These persons have been considered as having minimal clerical skills, and their job is to check the entrance/exit at the secured building entering the appropriate information into the system, without having any further access to data. Moreover, we don't want to let them access the data into CCCS (accidently or on purpose) by using the keyboard or the mouse. Therefore, the use of speech recognition system and bar coding:

- enables *guards* to run the prototype with out having to use any of the traditional input devices, such as keyboard or mouse.
- enables *guards* to enter information about entries/exits and check individual's records with just a bar code scanning.
- provides a higher security level. First by permitting only persons positively voice-recognized as *guards* to access the system and enter data. Second, by not allowing *guards* to access further data of CCCS (they don't have any keyboard or mouse).
- enables *guards* to quickly produce productive results with a minimum of training time.

The *Real Users* (U) and *Readers* (R) could also start the CCCS execution using speech recognition. But, because these users can access more sensitive data and procedures in the prototype, there is also a Log-in procedure for them to pass (it consists of a LogOn ID number and a password). After a positively recognition, they can gain access to CCCS data. This way is also available to all level users for the case of lack of

voice recognizer and bar code scanner (or if they are out of order), figure 3.1. This way offers the same advantages as previously. Moreover, it improves security by adding one more check point (i.e., the Log-in procedure).

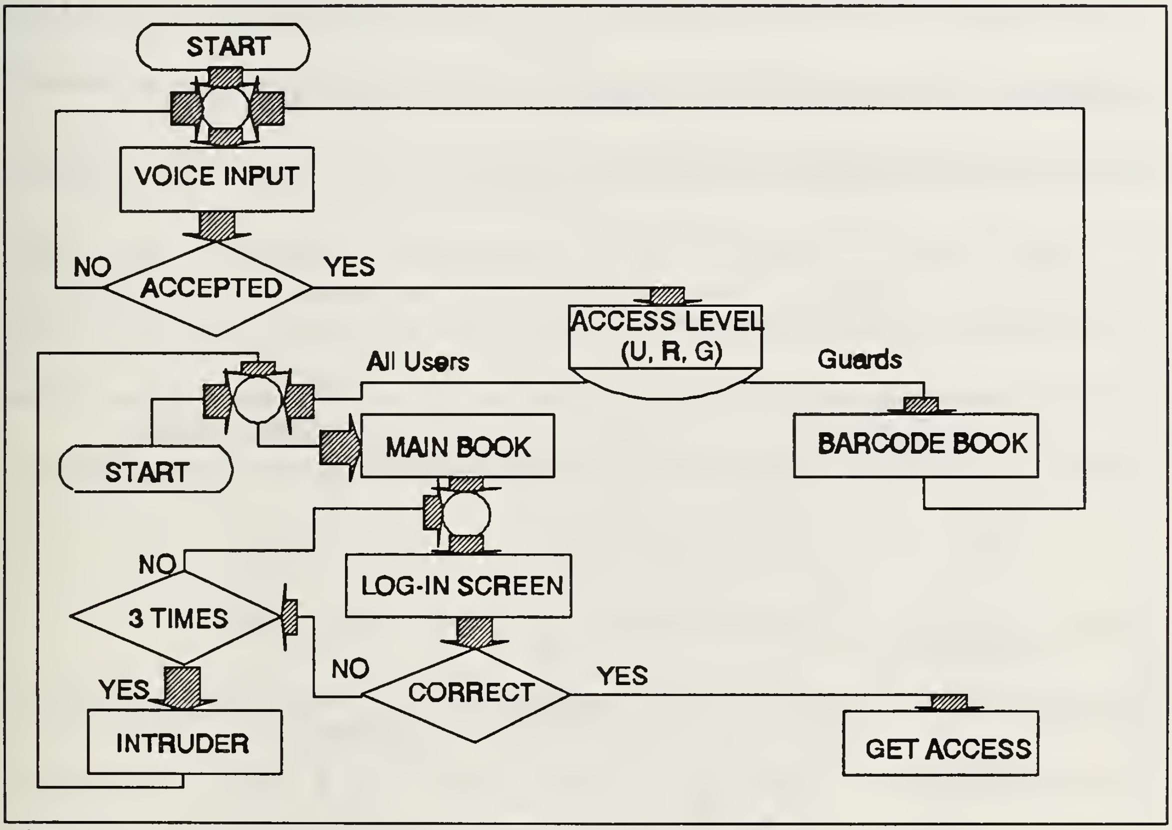

Figure 3.1 Logic Sequence Accessing CCCS Prototype

3. The Actual Database

The CCCS prototype is a multimedia database, and its related ER Diagram [ELMASRI89] is shown in figure 3.2. The PERSON tupple is associated with the records of individuals who are granted entrance rights to the secured building. The SYSTEM-USER tupple is related to the all level users previously mentioned. Since we can not

manipulate a photo as a simple attribute of PERSON tupple, we have to produce a distinct tupple for the person's photograph itself. Each PHOTO instance accompanies the respective individual of the PERSON instance. Finally, ***Persons*** can enter/exit the building and the ***System-users*** (all three levels) can register all those information. Besides the entry/exit data, the first level *System-user* can manipulate all the data into the CCCS multimedia database, and access any supported report or information (the last capability is provided to intermediate level *System-users* too).

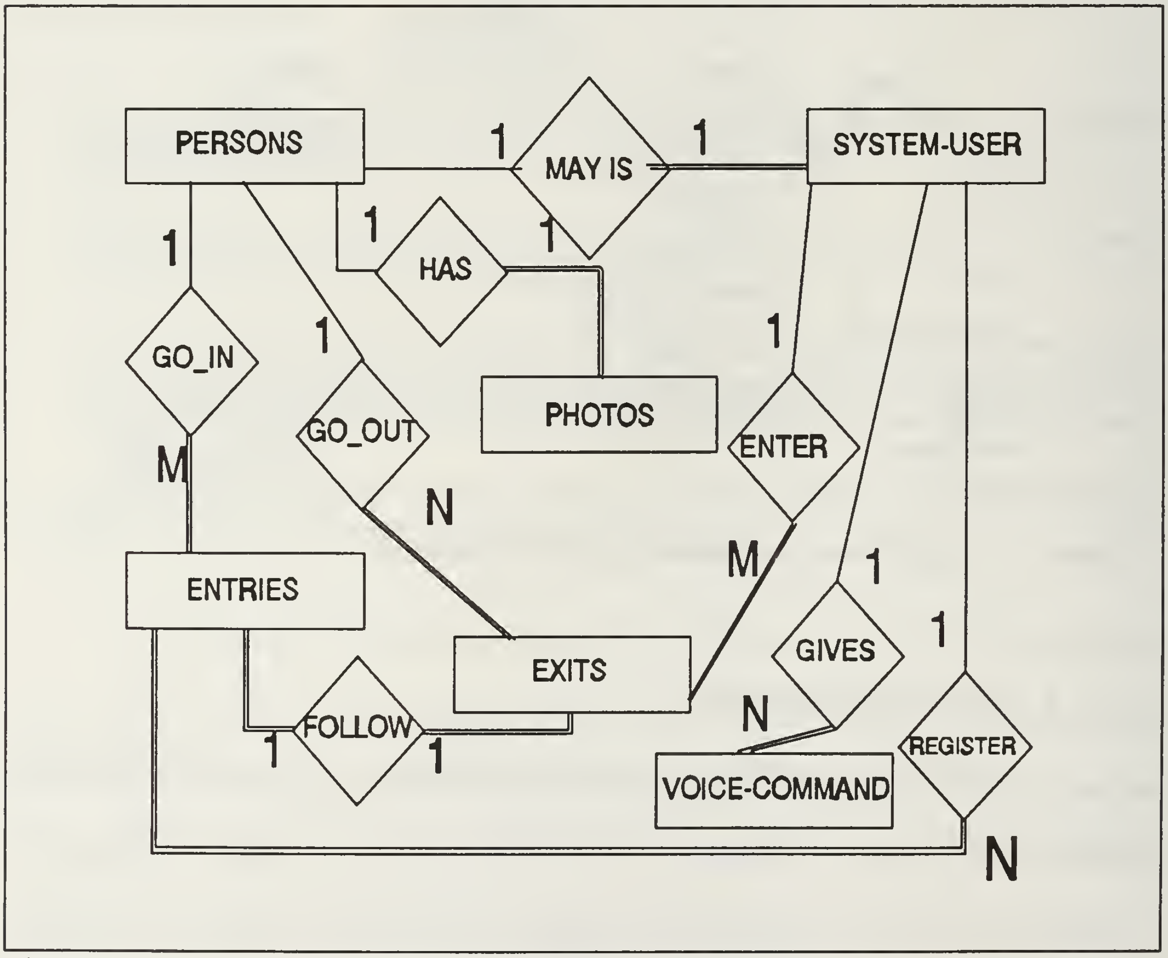

Figure 3.2 ER Diagram Of CCCS Multimedia Database

C. IMPLEMENTING THE PROTOTYPE

1. The Three Books Of CCCS

In order to accomplish the predefined goals, the CCCS prototype consists of three programs. Each of them is a distinct *book* in the terminology of Toolbook.

a. Main Book

The first book is called "**MAINBOOK.TBK**" and it can be accessed by all users (U,R and G categories). Its primary purpose is to be used by *Real Users* (U) and *Readers* (R) to manipulate and access data in CCCS. Secondarily, it enables users of the three categories to enter data about individual's entries/exits using the keyboard, mouse and bar code scanner (whether or not the last is available). It is allowed to be executed by using speech recognized commands, but users must also pass the log-in procedure.

b. Bar Code Book

The second *book*, called "**BARCODE.TBK**", is for the *guards* (G) to access the system and enter information about entries/exits using only voice recognition and bar code scanner. It is a run-time version of the part of the "MAINBOOK.TBK" which is related to bar coding. It allows users of G category to enter bar-coded ID numbers using the bar code scanner. This way, they verify the individual's right to access the secured building, check that the present person is the same one that referred in the database, and update the database of entries/exits. There is not a log-in procedure. Since the user can only enter information about entries/exits, without be able to access any other data into CCCS, the voice-recognized command to call the program is considered secure

enough. Besides, the *guards* don't have any other input devices except the voice-recognizer and the bar code scanner.

c. *Photo Book*

The last *book* is called "**PHOTO.TBK**". It could be considered as a part of the person's database that is used to store and manipulate all the individual's photographs in the PERSON database. This is how we exploit the feature of ToolBook to accept and manipulate scanned objects, in order to add person's photographs producing a multimedia database.

2. Additional Information

Besides the three *books*, we need other files to support all the operations the CCCS prototype performs. In order to manipulate the PERSONS, SYSTEM-USERS and ENTRIES databases, we exploit the feature of ToolBook to recognize/produce/update the format and manipulate the data in dBASE III PLUS files. This capability is provided via the TBKDB3.DLL library in ToolBook, which allows ToolBook applications to communicate and work with dBASE III and dBASE III PLUS files.

CCCS prototype has also to store intruder's attempts to break into the system and person's entries/exits history. To accomplish that task, the system uses the TBKFILE.DLL library that allows ToolBook to perform some DOS actions from a script.

Finally, we use the Dynamic Data Exchange (DDE) feature of the MS Windows environment to manipulate photograph concurrently with the other person's attributes. Dynamic Data Exchange is a protocol that permits applications running under

MS Windows to talk to one another. In such a manner, scripts in "MAINBOOK.TBK" and "BARCODE>TBK" send commands to, or get information from the "PHOTO.TBK".

D. CONSTRAINTS

1. Speech Recognition Systems

The voice recognition is difficult at best with present commercial technology for IBM compatibles. In present, the provided commercial systems are speaker-dependent isolated systems which function within DOS environment. Therefore, the voice recognized commands might be any command that can be recognized and executed by the DOS (i.e., DOS commands, batch files, executable files).

Since the CCCS prototype has been implemented in ToolBook which runs only under MS Windows environment, the starting of any of the previous referred *books* can easily be done using a voice recognized command. On the contrary, the ending (i.e., exiting CCCS) would leave the user within the MS Windows environment. Therefore, a new given voice-command will not be executed. In order to use the speech recognition systems, we need the CCCS prototype to be called from within DOS environment and after its completion to return control back to DOS.

2. System Hardware/Software Requirements

a. Microsoft Windows 3.0

The minimum software and hardware requirements a computer needs to run MS Windows 3.0 successfully, referred in [WINDOWS90], are:

- MS-DOS or PC-DOS, version 3.1 or later;
- At least *standard mode*, a personal computer with 80286 processor or higher and 1MB or more memory;
- A hard disk with 6 to 8 MB of free disk space, and at least one floppy disk drive;
- A monitor that is supported by MS Windows; and
- A mouse that is supported by MS Windows.

b. ToolBook

According [TOOLBOOK89], ToolBook has been designed to work with any Microsoft Windows-compatible PC with 80286 or higher processor. Furthermore, the minimum system requirements to run ToolBook are:

- A personal computer with a n 80286 or higher processor;
- Microsoft Windows version 3.0 or higher;
- DOS 3.1 or higher;
- 640KB plus 256KB extended memory (1.5MB memory recommended);
- One 1.2MB ($5^1/_4$") or 720KB ($3^1/_2$") disk drive;
- A hard disk with 2MB to 8MB of free disk space;
- A monitor and adapter card for VGA, EGA, or Hercules™ graphics; and
- A Windows-compatible mouse or other pointing device.

c. Scanner

A scanner to enter the photographs of the persons who are granted access rights to secured building (i.e., individuals in the PERSON database), is needed too. Moreover, since the acceptable format of scanned objects within MS Windows environment is bitmap, we also need a software package to convert the scanned images, from whatever format is, to bitmap.

IV. CONTROL COMMAND CHECK SYSTEM

A. INTRODUCTION

1. Preface

The **Control Command Check System (CCCS)** is a multimedia database prototype system. It has the capability to use speech recognition, a bar code scanner, and a database to store information (traditional data plus image) about an individual. An organization that must control access can use this system to verify authentication.

CCCS has three access levels. The highest access level is *Real Users* (U level), whose primary job is to maintain the access records in the database. They also have access to all operations and data provided by the CCCS (i.e., they are *full users* of CCCS). The intermediate access level is *Readers* (R level). *Readers* are granted *read*-only access, not *write* access. That is, they can access information on an individual in the database, but they can not directly access user's record and they can not modify anything in CCCS database. Finally, the third level is *Guards* (G level); *Guards* perform entry level tasks; such as checking ID's for Entry and Exit to an organizarion. Additionally, they store the entry/exit information in the appropriate files of CCCS.

There are three *books* that can be accessed, each for different reasons. The first book, *MAINBOOK.TBK*, is used by *Real Users* and *Readers*. It also allows all level users to enter entry/exit information. The second book, *BARCODE.TBK*, is used by guards to check individuals for access to an entry/exit point, and then store this information into the

system. The third book, *PHOTO.TBK*, is used as a database to store individual's images, and has the capability to manipulate that kind of data.

When an individual wants access to the building, he/she presents a card with a unique bar code on it to the guard. If a speech recognition system is available, the guard enters a voice recognized command. This command executes a DOS batch file which activates an MS Windows environment, ToolBook, and loads the "BARCODE.TBK" book. The guard uses the scanner to read the bar code on an individual's ID card. Then, the CCCS responses by displaying the individual's record. If the ID number is valid, information of entry/exit is stored in the respective database. However, if the ID number is invalid, the CCCS sends a failure message back to the user.

If a speech recognizer or a bar code scanner is not available, the data has to be entered using a keyboard and a mouse. In this case, a voice recognized command might be entered, but the loaded book will be the "MainBook.tbk" which contains a log-in procedure to allow further access, depending on the user's access level (U, R, G).

2. Additional Information

The following section, *Fundamentals*, demonstrates how to work with the CCCS. It guides the reader navigating via the different screens of the program, and explains the displayed menu or submenu functions. Moreover, it describes the features of the prototype and what the system requires or accomplishes during each procedure. The preliminary work, needed for the manipulation of a person's photograph in the database, is also described.

Finally, the last section, *Objectives and Goals*, examines the CCCS prototype system to determine if it has accomplished the original objectives.

B. FUNDAMENTALS

The description of the CCCS prototype includes the descriptions of how to work with each one of its three *books*: MainBook, BarCode, and Photo. The descriptions will be done via the provided screens of the specific book.

1. MainBook

a. Log-On Into MainBook

After the "MainBook.tbk" has been loaded by a voice recognized command or not, the screen shown in figure 4.1 is displayed. The system displays the title and the possible combined means in the multimedia database, and offers two choices to the user: to log-on into the system, or to exit going back to Windows or DOS environment. If the user selects the "EXIT" button, then the system will ask for a verification and if the user confirms that it will exit.

Figure 4.2 presents the starting of the log-in procedure. First, the system asks the user to enter his/her logon ID. Then, it requests the password. If the entered combination does not exist in the user's database, the system starts the log-in procedure again. If the user fails to enter a correct combination of ID and password three times, the system considers that as an intruder's attempt, stores the ID and password, and returns to first page (figure 4.1).

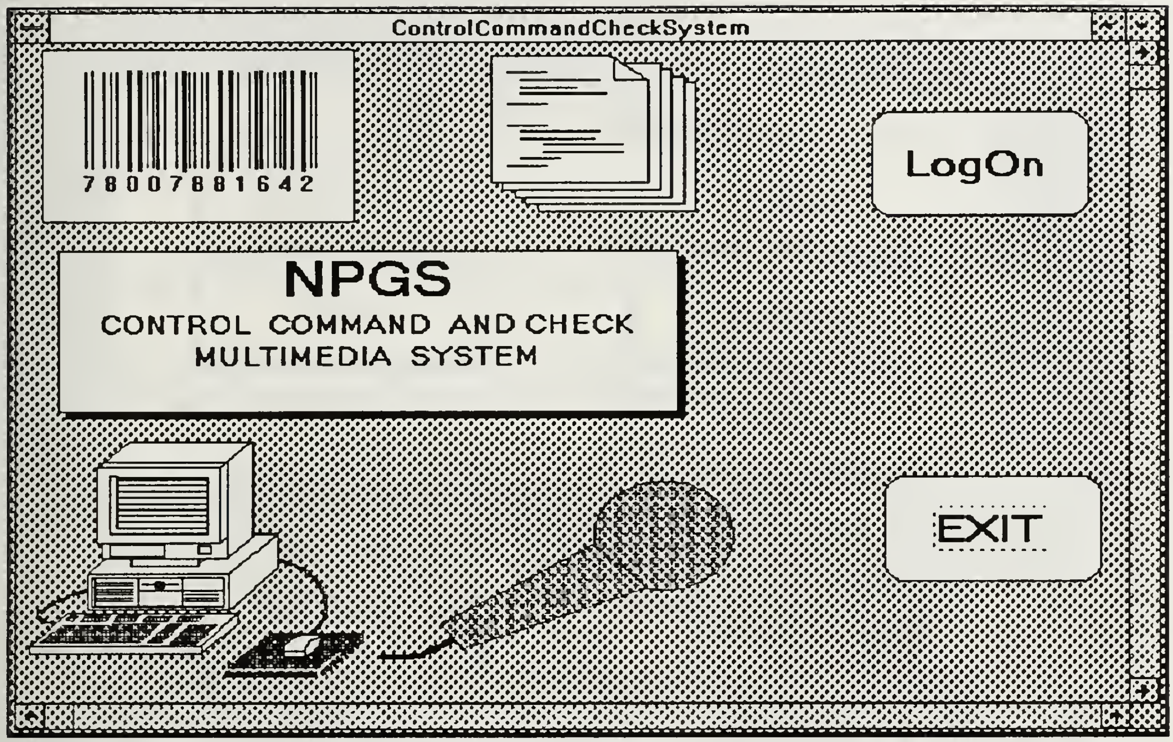

Figure 4.1 First Screen of MainBook in CCCS

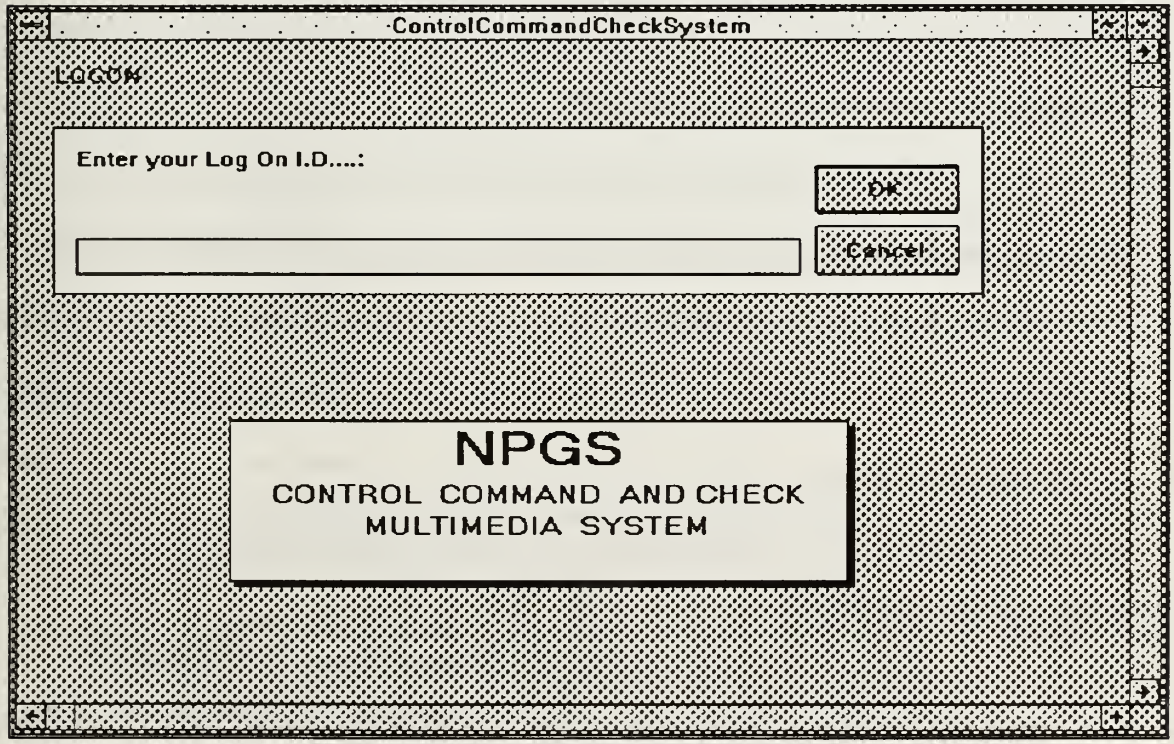

Figure 4.2 The Log-In Procedure of MainBook in CCCS

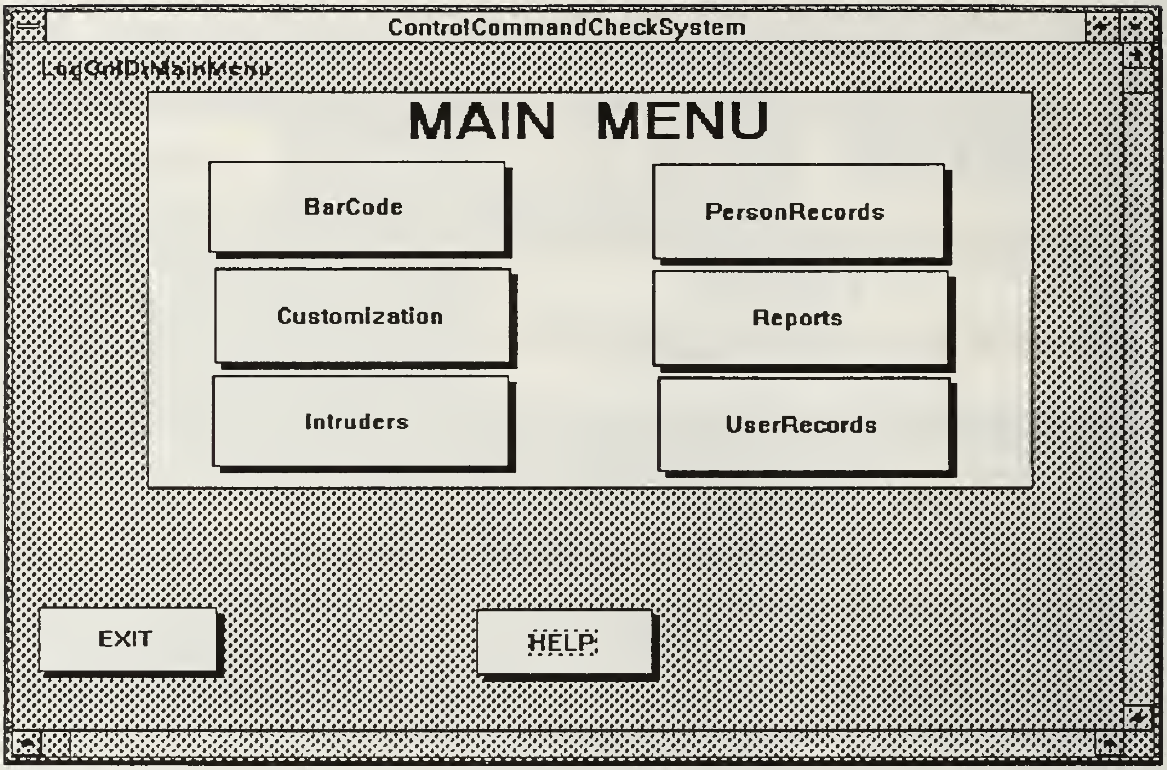

Figure 4.3 Main Menu For Full Users Of MainBook In CCCS

b. Main Menu For Real Users

After a user has been recognized by the system as a U-level user (full-access users), the system grants him/her access to the main menu shown in figure 4.3. There are two buttons at the bottom of the screen: one is the "EXIT" button for exiting the *MainBook*, and the other is the "HELP" button which is an on-screen help explaining all the buttons and their related functions. At the upper left-hand corner of the screen the current path appears. Finally, at the middle, the main menu provides six selection buttons to the user:

- **BarCode** selection enables user to check an individual who is asking for entry/exit access, and store the specific entry/exit information;

- **Customization** selection allows user to select the colors to be used for background, foreground and text for personal satisfaction;

- **Intruders** selection informs user if any intruding attempts to break into the system have been occurred;

- **PersonRecords** selection transfers control to a menu for manipulating the database records of the persons who are allowed access to secured building;

- **Reports** selection enables user to read/print reports about persons who currently or in the past have entered the secured building; and

- **UserRecords** selection transfers control to a menu for manipulating the database records of all system-users (categories U, R and G).

c. Main Menu for Readers

Whether a user has been recognized by the system as a R-level user (read-access users), the system grants him/her access to the main menu shown in figure 4.4. There are two buttons at the bottom of the screen: one is the "EXIT" button for exiting the *MainBook*, and the other is the "HELP" button which is an on-screen help explaining all the buttons and their related functions. At the upper left-hand corner of the screen the current path appears. Finally, at the middle, the main menu provides four selection buttons to the user:

- **BarCode** selection enables user to check an individual who is asking for entry/exit access, and store the specific entry/exit information;

- **Customization** selection allows user to select the colors to be used for background, foreground and text for personal satisfaction;

- **PersonRecords** selection transfers control to a menu for accessing information about database records of persons who are allowed access to secured building (it is a read right only); and

- **Reports** selection enables user to read/print reports about persons who currently or in the past have entered the secured building.

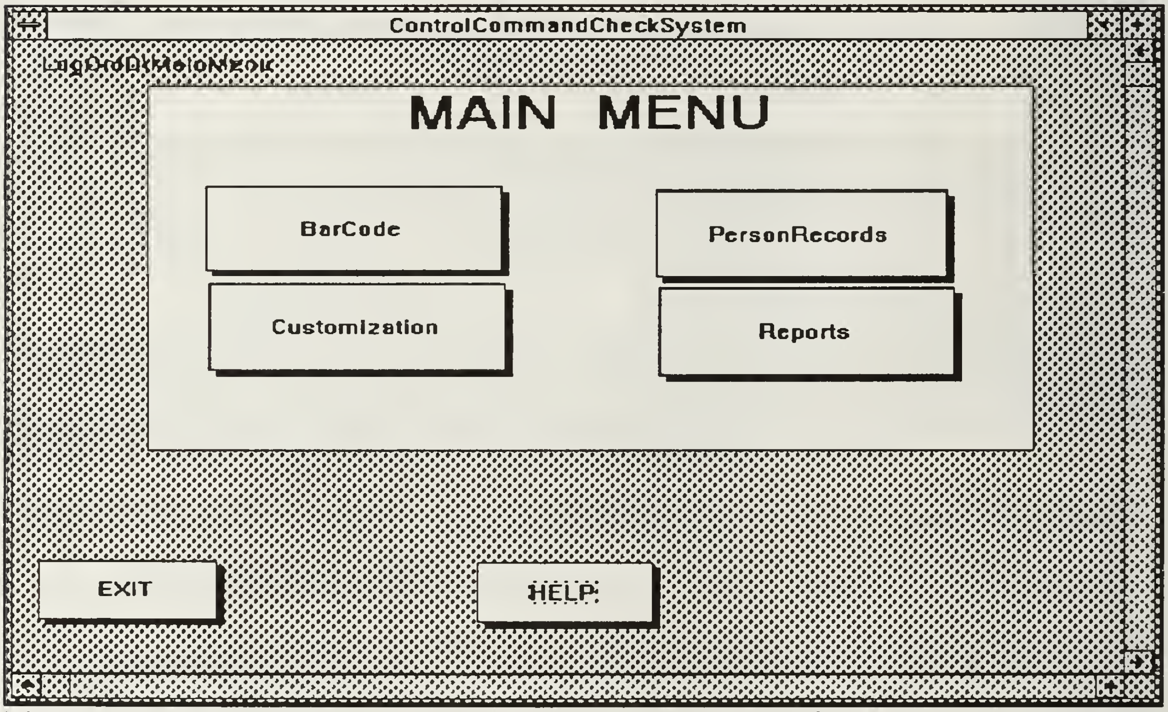

Figure 4.4 Main Menu For Readers in MainBook of CCCS

d. Main Menu For Guards

Whether a user has been recognized by the system as a G-level user (guards), the system grants him/her access to the main menu shown in figure 4.5. There are two buttons at the bottom of the screen: one is the "EXIT" button for exiting the *MainBooK*, and the other is the "HELP" button which is an on-screen help explaining all the buttons and their related functions. At the upper left-hand corner of the screen the

current path appears. Finally, at the middle, the main menu provides the user with only one choice:

- **BarCode** selection enables user to check an individual who is asking for entry/exit access, and store the specific entry/exit information.

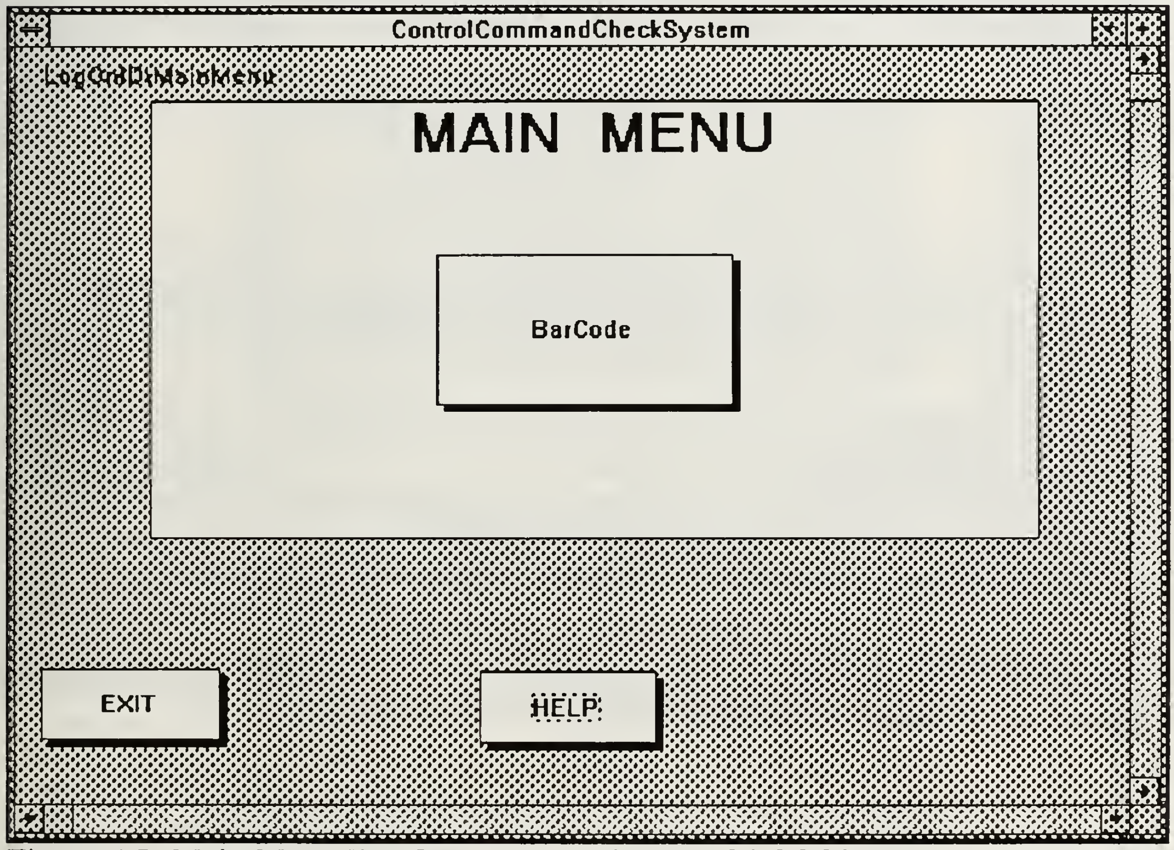

Figure 4.5 Main Menu For Guards In MainBook Of CCCS

e. Accessing On-Screen Help

As mentioned previously, there is a "HELP" button in all screens with buttons that perform some functions. To get the on-screen help, the user must simply click on the "HELP" button. The system, then, will display a text field which is a detailed

description of the function of each button on the screen (see figure 4.6). In help screens there is only one button. Clicking on that button, the user can return to the previous menu.

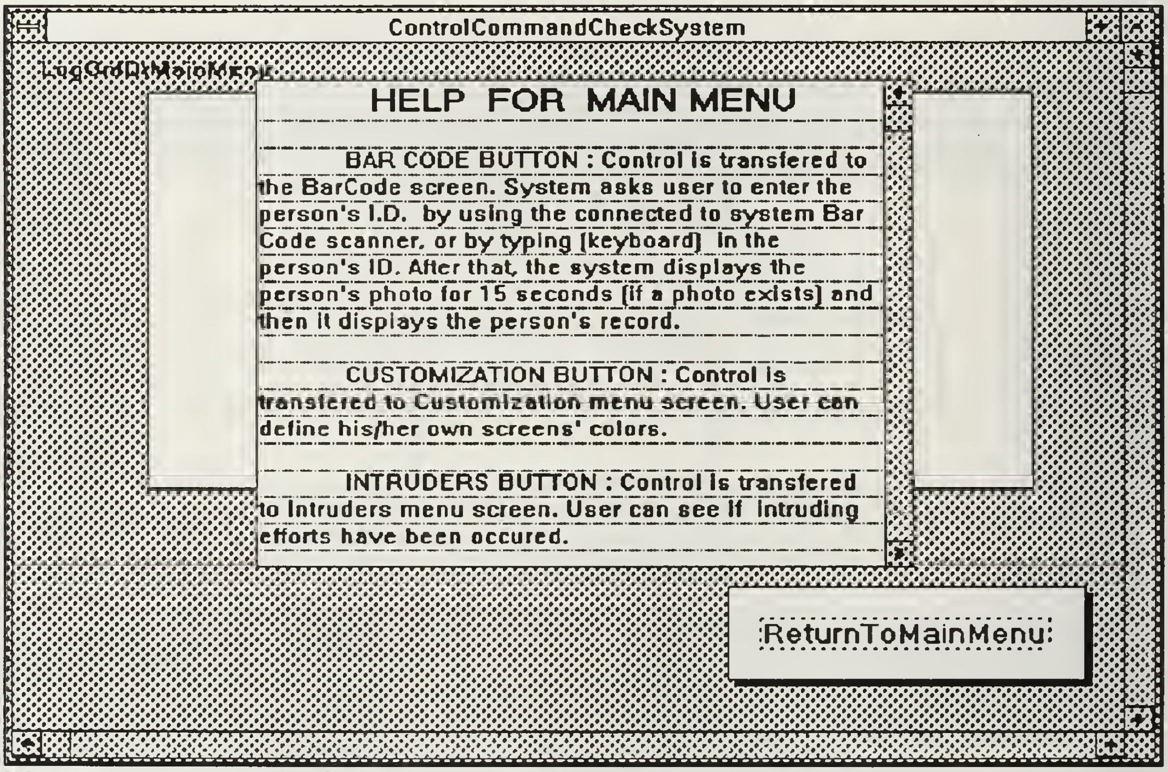

Figure 4.6 **Accessing On-Screen Help In MainBook Of CCCS**

f. *Entering Bar Codes*

As mentioned before, the bar code selection appears in all user menus. After the user selects the "BarCode" button from the main menu, the system displays the screen of figure 4.7. First, at the upper left-hand corner of the screen the current path is shown. Second, the fields of a person's record are displayed. Finally, the system requests a person's ID to be entered. The person's ID number can be entered using the keyboard

(if a scanner is not available). The *MainBook* supports entering bar code ID numbers if a bar code scanner is not available. Otherwise the *BARCODE.TBK* must be used. In this case, the system enables a user to type the ID number, to check that what he/she has typed is correct and then to enter that ID number into the system by either clicking on the "OK" button or by pressing the "Enter" key. If an error occurs or the user wants to cancel the procedure, he/she can quit by clicking on the "Cancel" button returning to the respective main menu depending on his/her access level (U, R, or G).

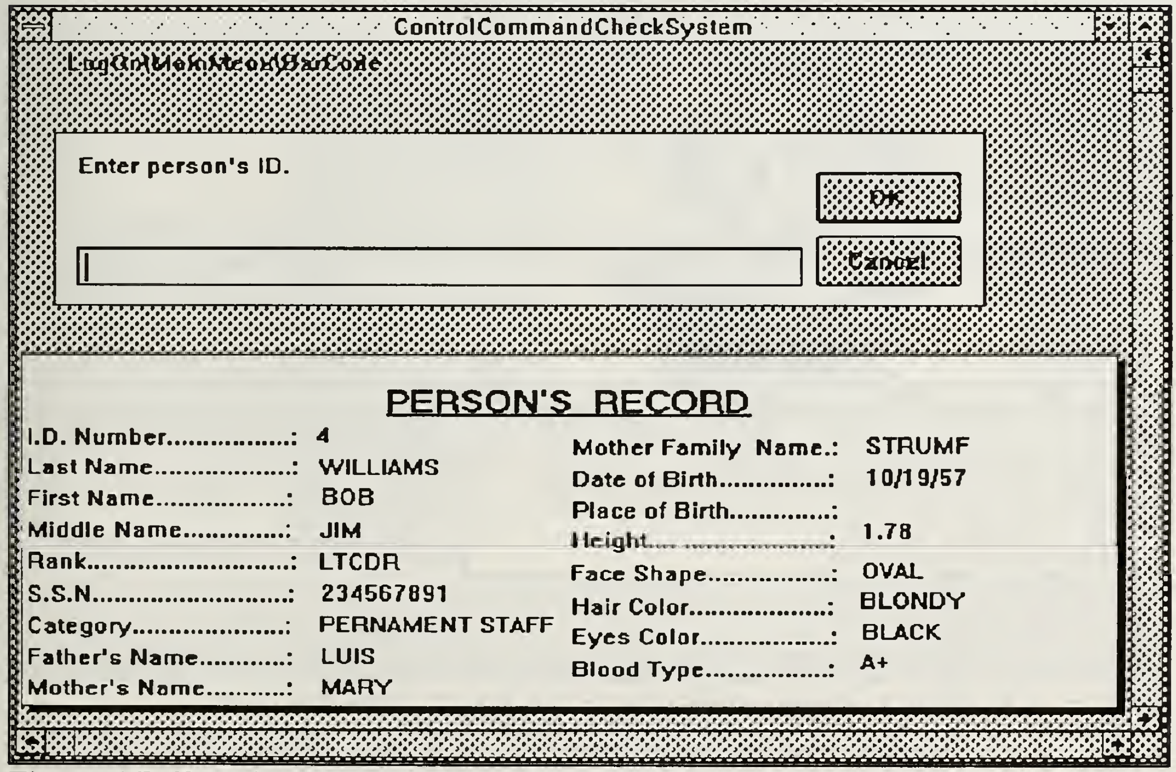

Figure 4.7 Entering Bar Code ID Into MainBook of CCCS

When a user enters the individual's ID number into the system, the system tries to find the specified record in the PERSONS database. If the search fails, the

system will inform the user displaying a failure message, such as given ID number not found, system error, etc. Otherwise, the search locates the record related to passed-in ID number. Then the system opens and searches the photographs database to find the respective person's image. If there is not one, it will display the screen of figure 4.8 which is the first page of the *PHOTO.TBK* for 15 seconds. If an image is found, the system will display that person's image for 15 seconds (see figure 4.9), allowing a user to compare it with the owner of ID card.

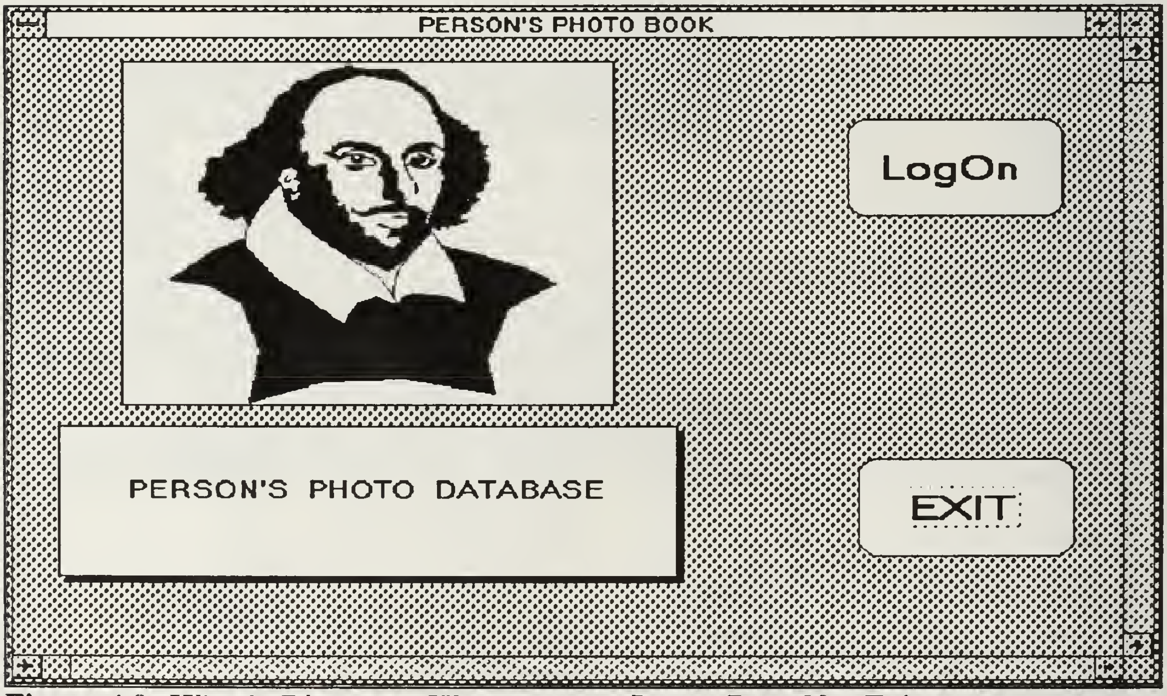

Figure 4.8 What's Displayed Whenever An Image Does Not Exist

After displaying the individual's image 15 seconds, the system returns to the bar code screen. The record field contain the information of the specified person. The system also informs the user, see figure 4.10, if the person is entering or is exiting, and

stores that information into entries/exits database respectively. The user clicks on the "OK" button to exit the bar code procedure and return to main menu.

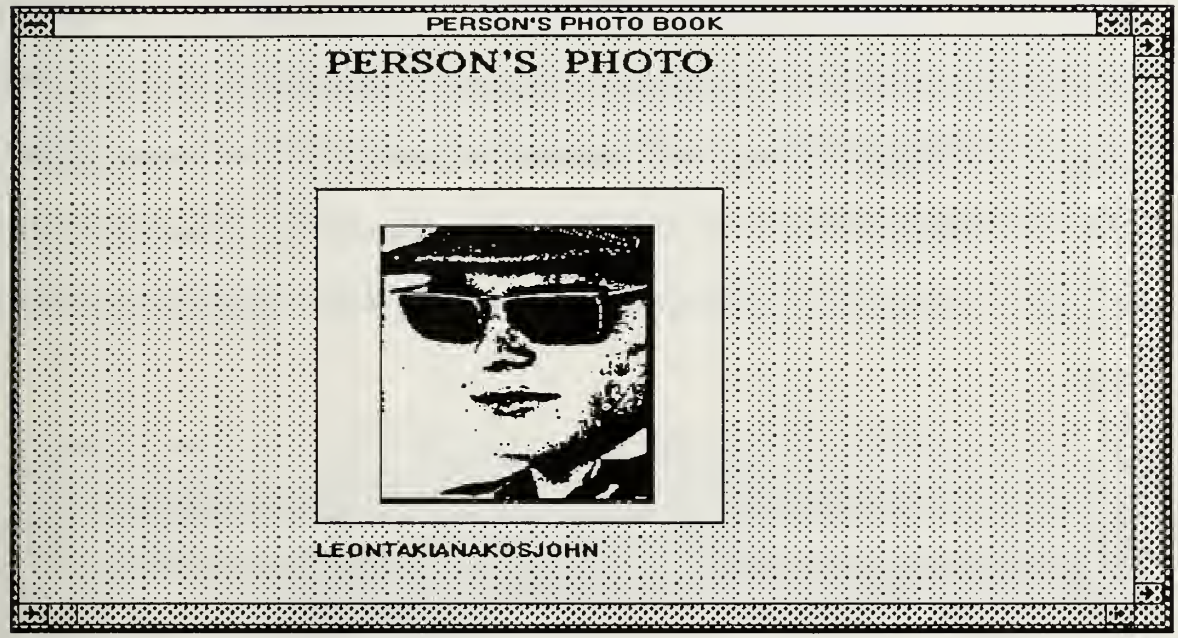

Figure 4.9 A Person's Image

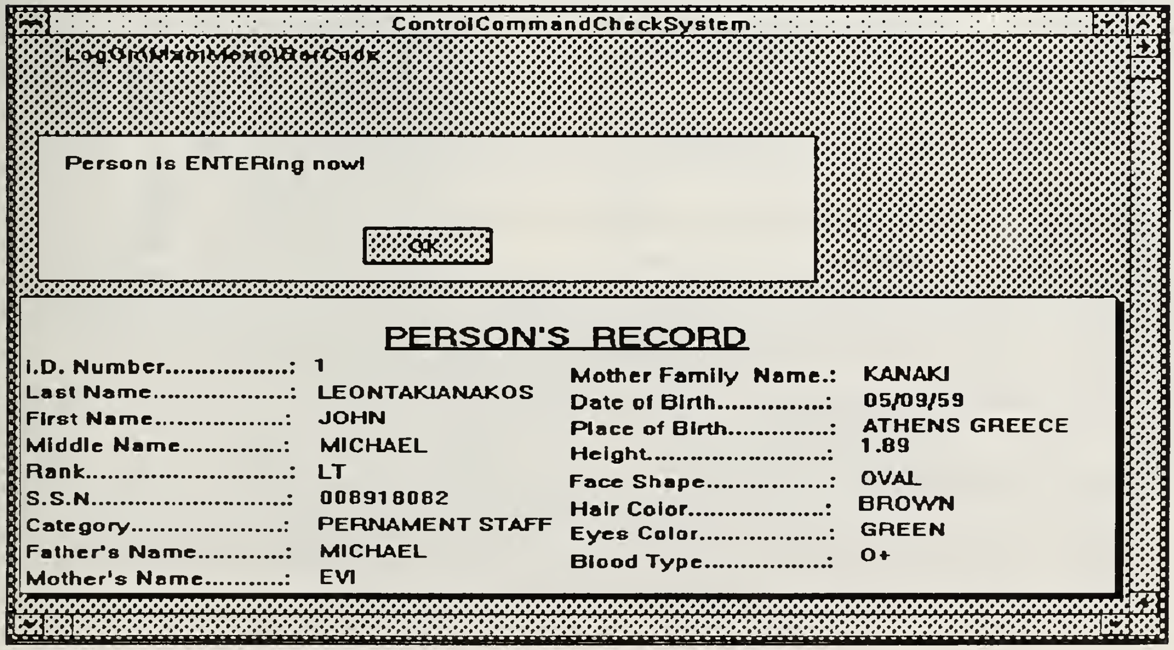

Figure 4.10 System Informs User About Person Entry/Exit

g. Customizing The Screen

The CCCS prototype provides U and R users with the capability to build their own colored screens to their satisfaction. After the user clicks on the "Customization" button, the screen of figure 4.11 is displayed. It is the screen customization menu. As in all screens, in the upper left-hand corner the current path is shown. At the bottom of the screen, there are the general purpose buttons "EXIT", "HELP", "DefaultColor" and "OK". The first two buttons perform similar functions as previously described. The "DefaultColor" button is for getting back the system default colors.

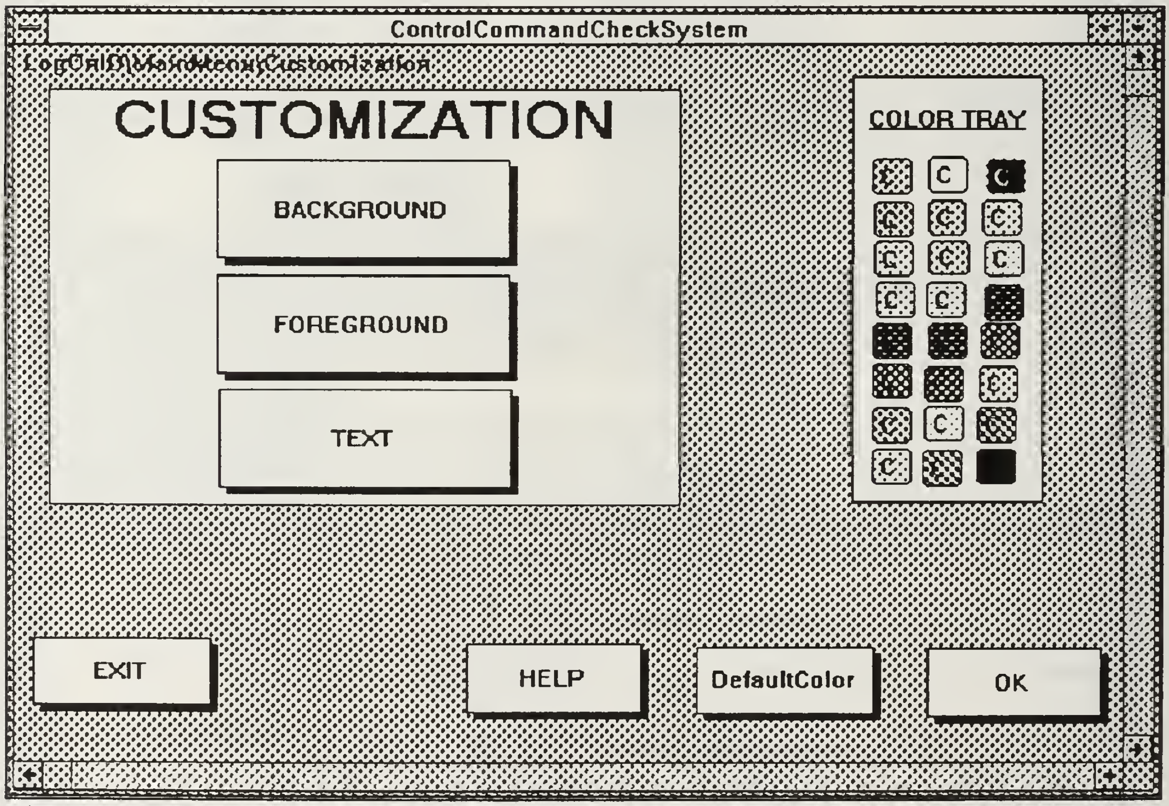

Figure 4.11 Customization Menu In MainBook Of CCCS

A detailed description of the screen customization procedure is provided via the on-screen help, see figure 4.12. The user must first select the recolored item (background, foreground or text). Then by clicking on the color buttons in the palette, he/she can watch the selected item change color. The user must verify his/her final color selection by clicking on the "OK" button. After the user has built the color screen, by defining colors to the background, foreground and text, the user verifies final selection by clicking on the "OK" button, which recolors all the screens in the *MAINBOOK.TBK*. Then, the user returns to the main menu screen.

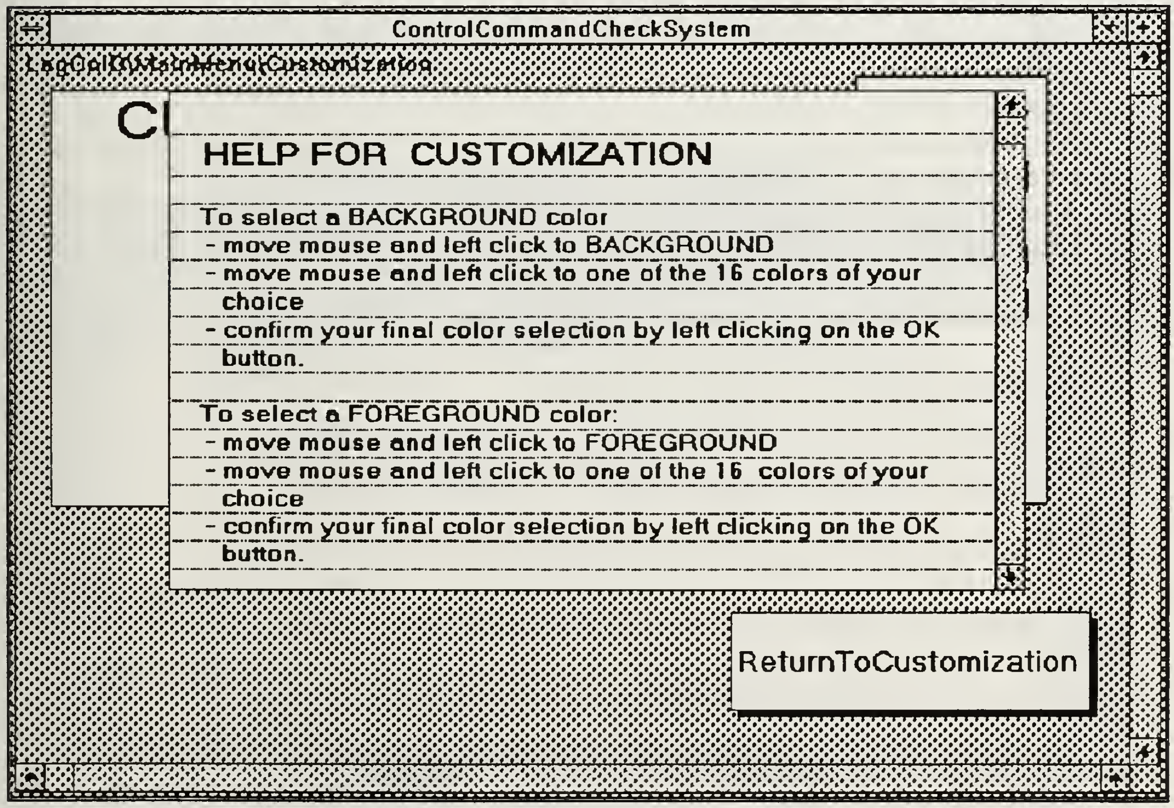

Figure 4.12 On-Screen Help For Screen Customization

h. Keeping Track Of Intruder Attempts

After clicking on the "Intruders" button in the main menu, the full-user gets the screen of figure 4.13. The current path is shown, as always, in the upper left-hand corner. The text field contains information about attempts to break into the system. The provided information are: date, time, entered user's ID number and password. The system enables a user to :

- print the displayed report by clicking on the "PrintIntruders" button;
- delete the file where the displayed intruders data is stored, by clicking on the "DelIntruderFile" button; and
- return back to the main menu, by clicking on the "ReturnToMainMenu" button.

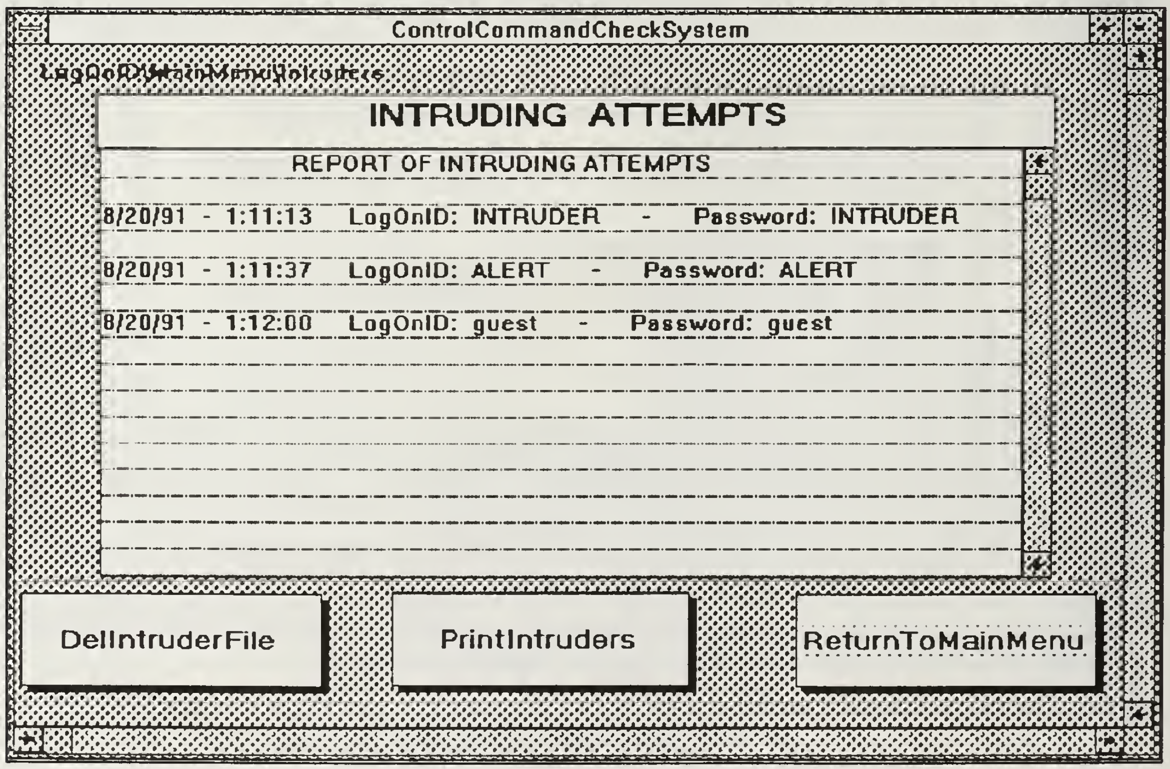

Figure 4.13 Intruder's Attempts Screen In MainBook Of CCCS

Finally, if an intruding attempts file has not been created by the system, an appropriate message informs the user that intruder's file doesn't exist, see figure 4.14

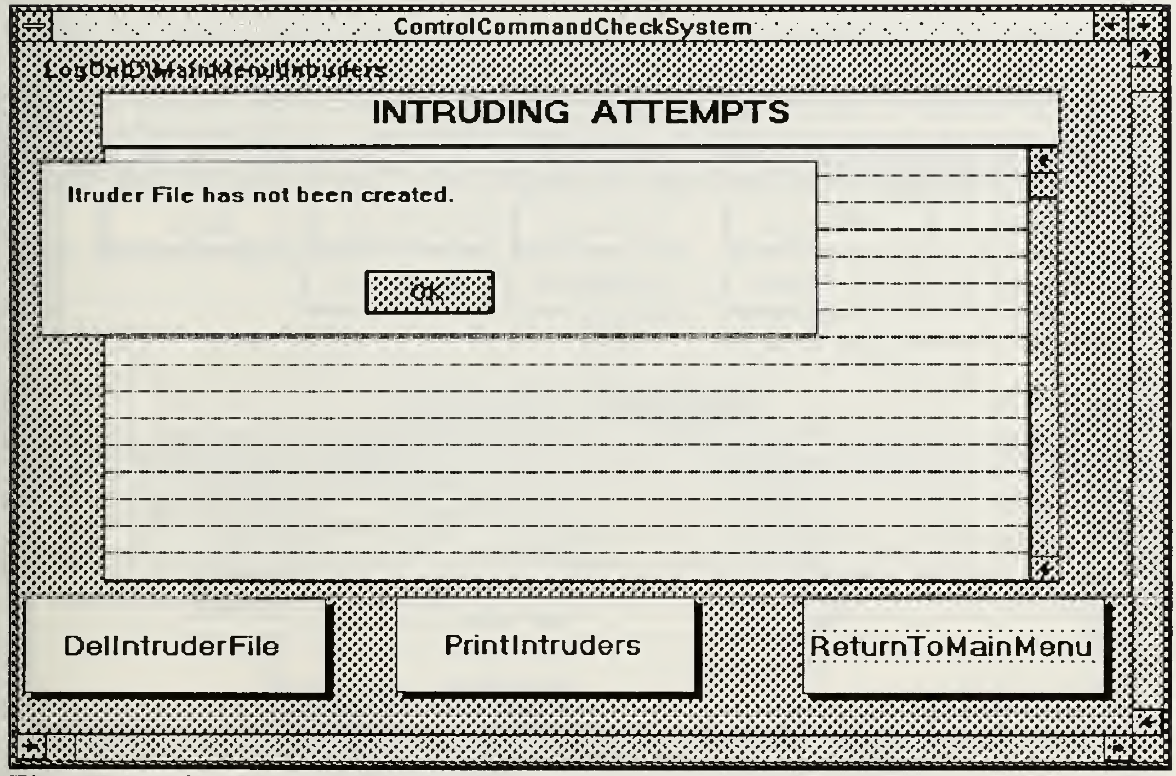

Figure 4.14 System Informs "No Intruders"

i. Manipulating Persons Database

The CCCS provides write/read access rights to full-users. By clicking on the "PersonRecords" button at the main menu, a *Real User* gets the screen shown in figure 4.15. The current path, in the upper left-hand corner, informs the user that this is the third level in the hierarchy. At the bottom of the screen, three general purpose buttons exist: to exit the entire *book* click on the "EXIT" button, to get on-screen help click on the "HELP" button, and to return back to the main menu click on the

"ReturnToMainMenu" button. When the "ReturnToMainMenu" button is clicked, the system asks for the user's verification, in order to quit the menu of person's database and return to the main menu.

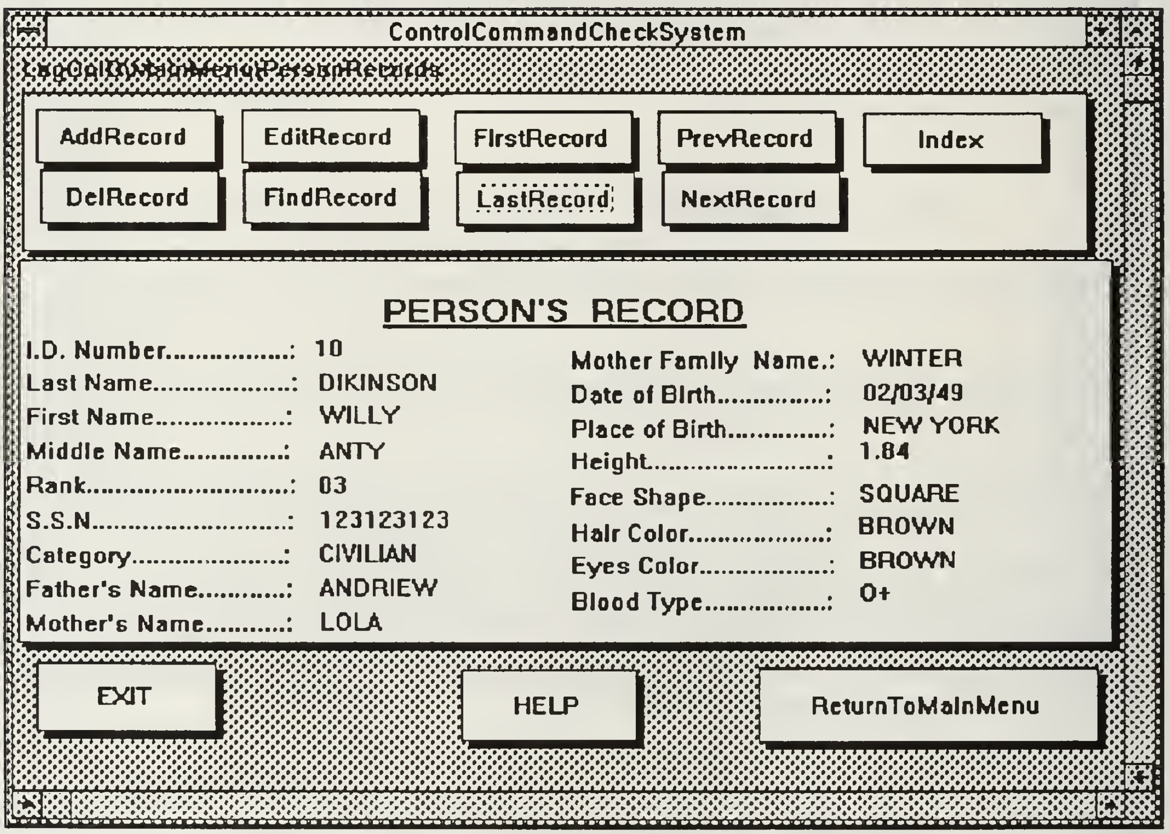

Figure 4.15 Persons Database Manipulation Menu in MainBook

The person's record, at the middle of the screen, contains all the data related to PERSONS tupple, except the photograph. Above the person's record, there is a menu to manipulate the person's database. The menu incudes nine categories which perform specific functions. Each function is described in details in the following sections.

(1) Add A New Record. The "AddRecord" procedure is a loop, i.e., the system asks the user, see figure 4.16, if he/she wants to add a new record, until he/she wants to stop. If the user's response is affirmative (i.e., he/she wants to add a new record), the system automatically creates a new record ID number. Then, it requests data to fill the person's record fields as shown in figure 4.17. If the user's response is negative (he/she does not want to add a new record), the system exits the loop and returns to the screen shown in figure 4.15.

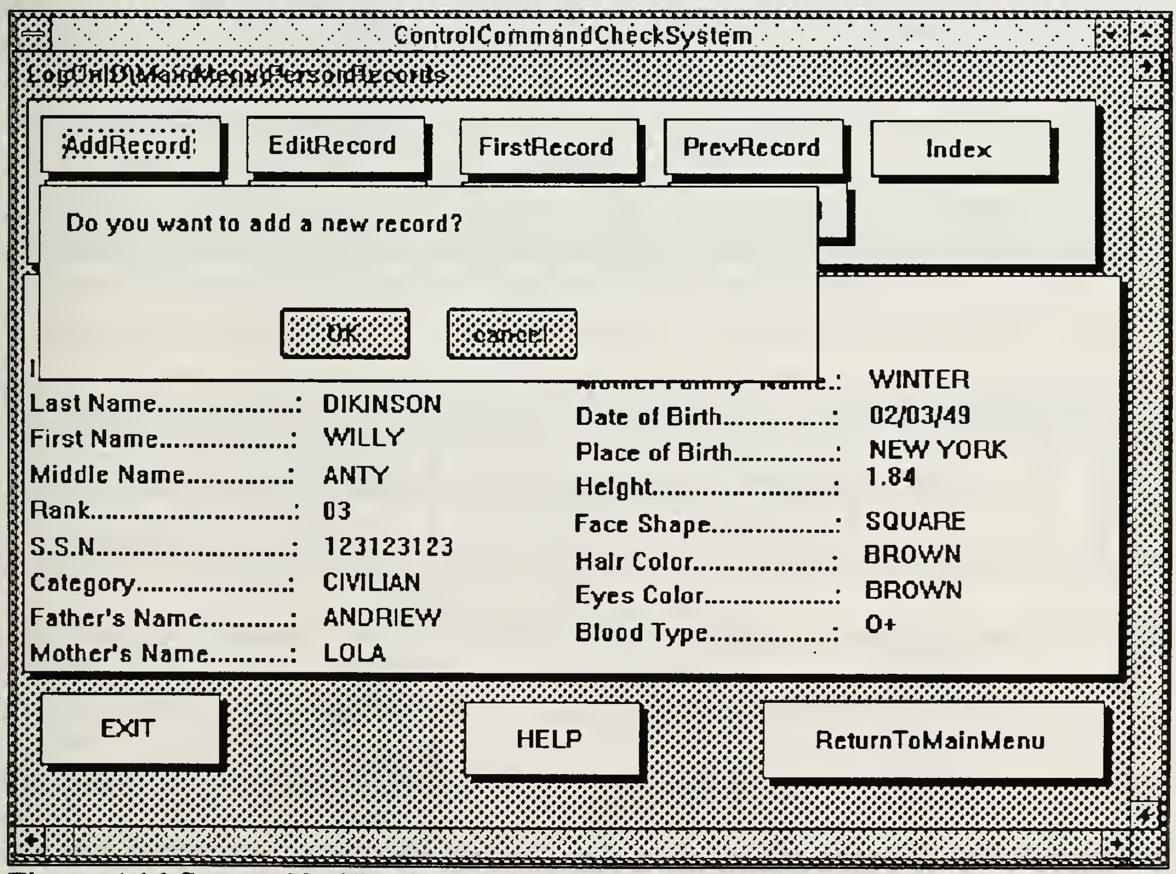

Figure 4.16 System Verification For Adding A New Person

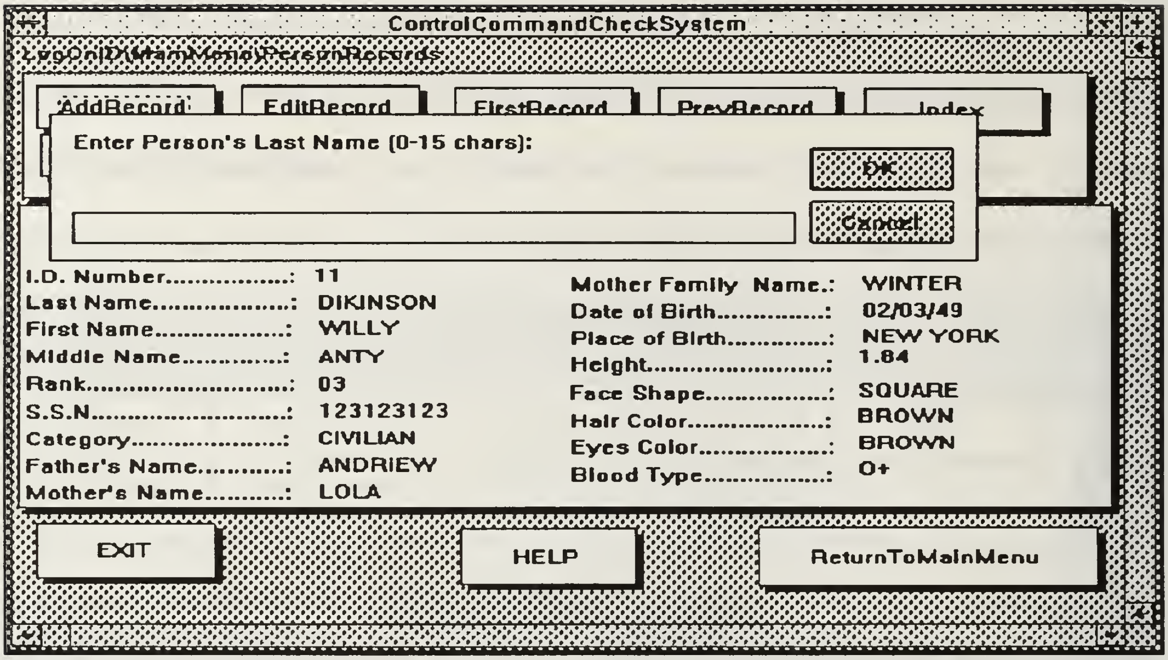

Figure 4.17 System Requests For Person's Last Name

(2) Delete An Existing Record. The "DelRecord" button deletes the currently displayed person's record. To delete a specific record, first, the user must find it in the "FindRecord" procedure. The system always verifies the deletion of a record as shown in figure 4.18.

(3) Edit An Existing Record. The "EditRecord" button allows a user to modify the currently displayed record. To edit a specific record, the user must find it in the "FindRecord" procedure, and then click on the "EditRecord" button. First, the system requests verification. Then, it informs the user that he/she must use the mouse and the keyboard to update the record fields on the screen. After that, it hides all the buttons and it displays the screen shown in figure 4.19. The user makes the corrections, and then

clicks on the "OK" button. If inappropriate input has been given in a field, the system will ask for correct input.

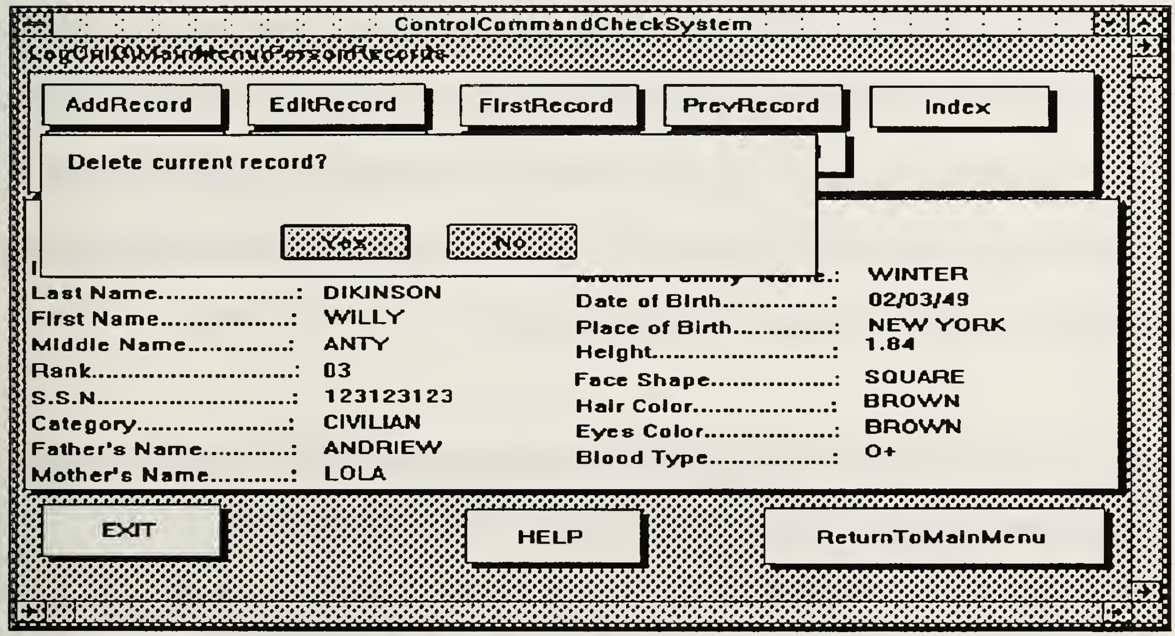

Figure 4.18 System Verifies The Deletion Of Current Record

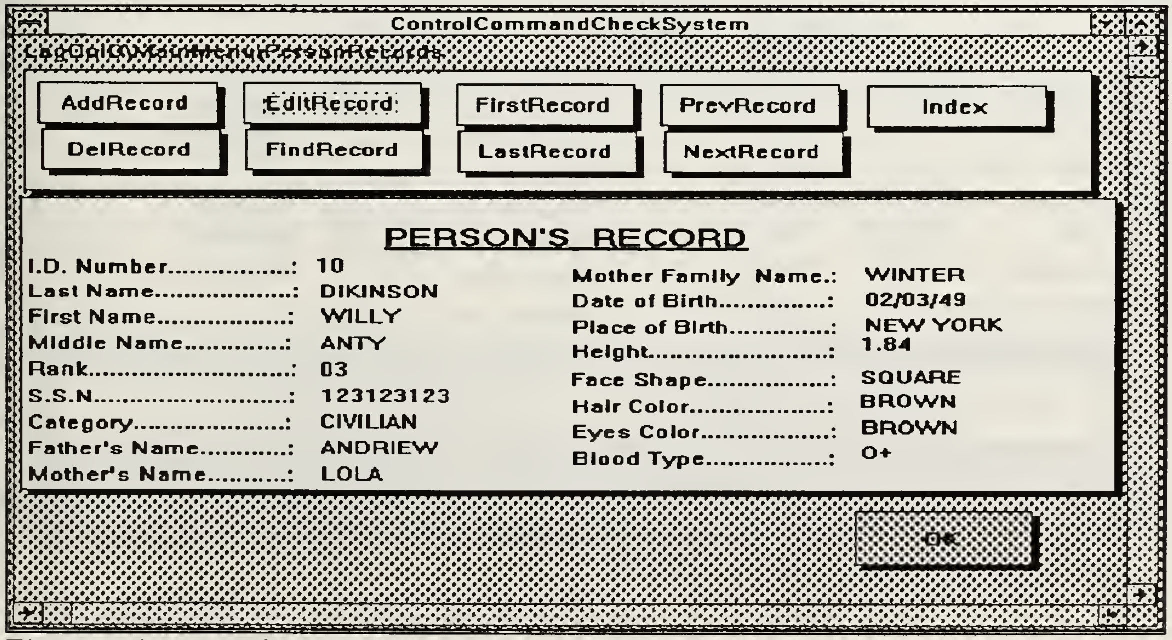

Figure 4.19 Updating A record In MainBook of CCCS

(4) Find An Existing Record. The "FindRecord" procedure enables a user to locate and display an existing record related to the entered *key*. This procedure interacts with the currently active index (person's ID number, SSN, or last name) as shown in figure 4.20. If no index is activated, the system asks user to activate an index and then proceed with the "FindRecord" procedure. If an index is active, the system asks user to enter the related key for the record to be found. If the entered key does not exist, the system displays the closest smaller number for ID number and SSN indexes, and it remains at the current record for last name index.

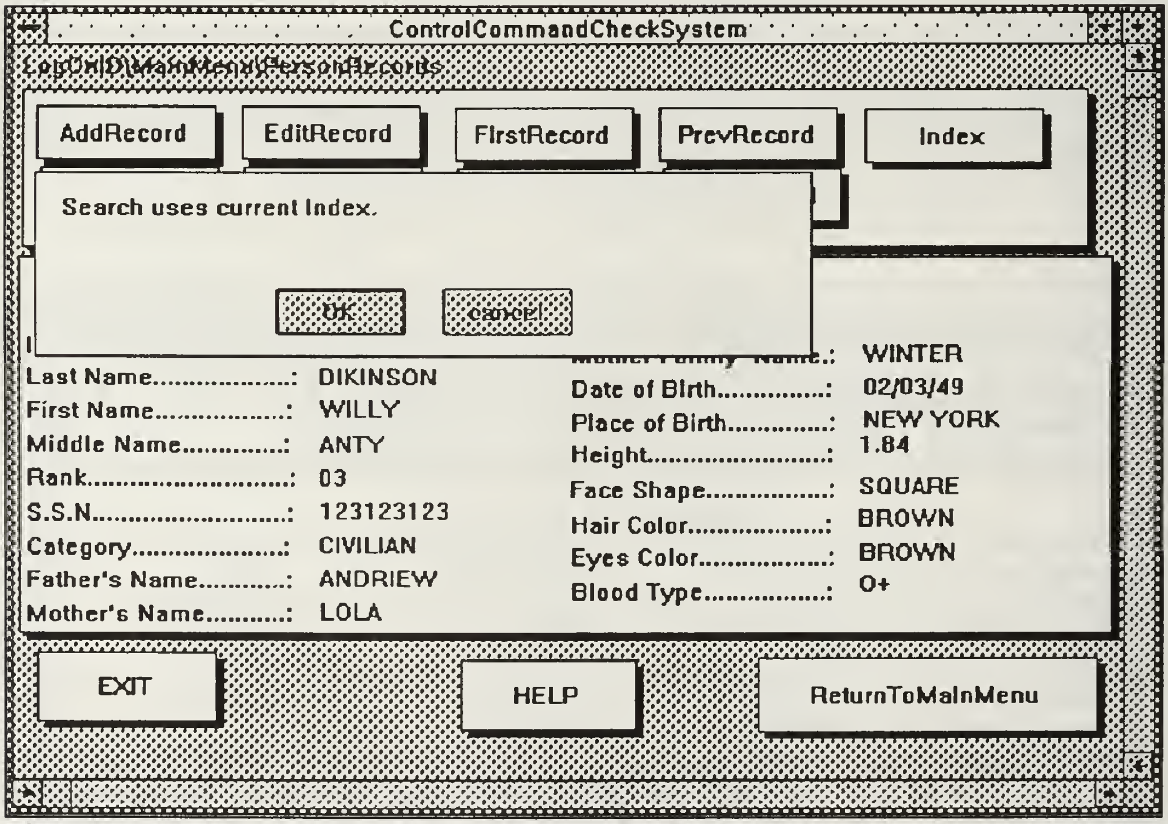

Figure 4.20 Finding A Person's Record

(5) *Accessing Records.* The buttons "LastRecord", "FirstRecord", "NextRecord" and "PreviousRecord" provides the capability of accessing records according to the active index and to their location. Therefore, last or first record is considered the respective last or first key in the active index file. Similarly, the next or previous record is the respective next or previous key of the currently displayed record. If the first or the last key is currently displayed, trying to get the previous or the next key respectively will result in a message similar to the one shown in figure 4.21. If no index is active, the system accesses records according to their physical storage in the database.

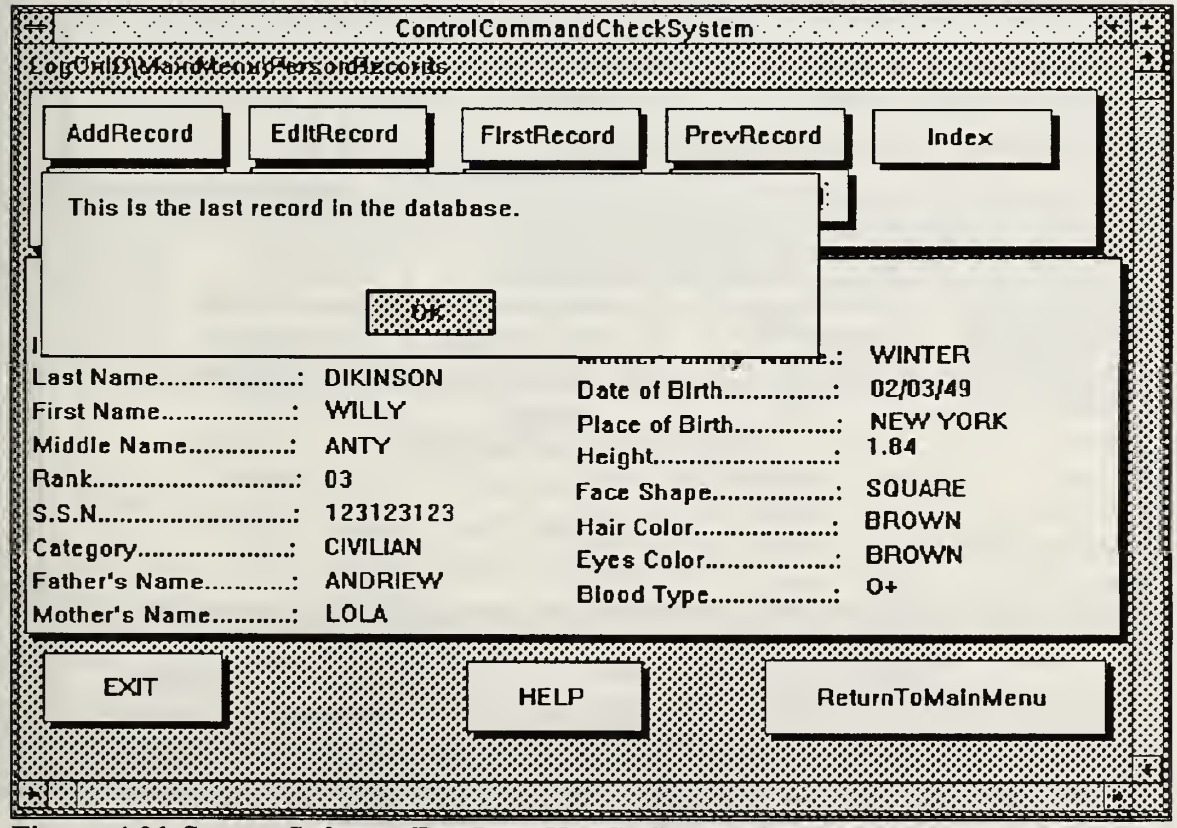

Figure 4.21 System Informs For Last Key In Index File

(6) *Indexing*. The system supports three different indexes: person's ID number, Social Security Number (SSN), and last name as shown in figure 4.22. The first two indexes are unique keys for each person. The ID number key is created and maintained by the system for each person, SSN is also a unique key, and last name indexing is not unique. Each time the user wants to activate an index (if no index is active) or to change to another index, he/she clicks on the "Index" button and selects the desirable index.

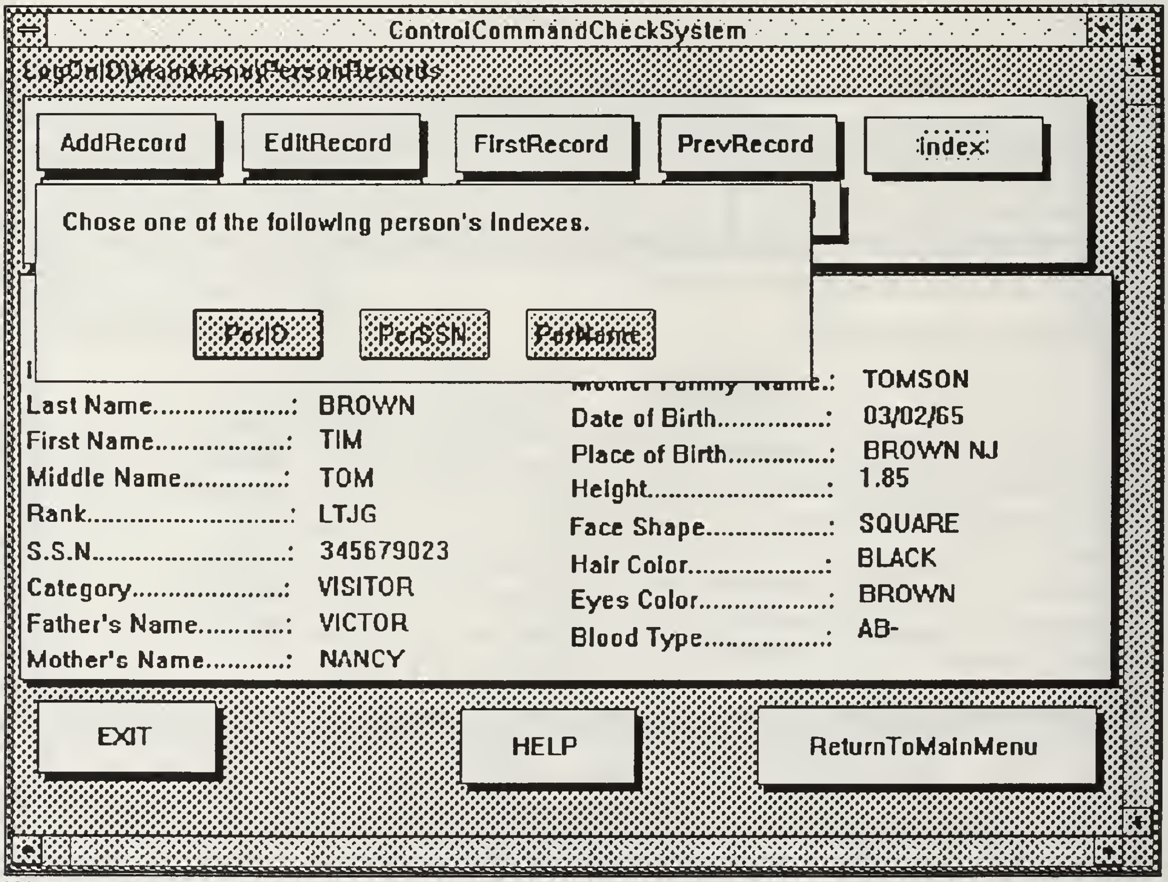

Figure 4.22 Indexes For Person's Records In MainBook of CCCS

j. Reading Persons Database

The CCCS provides read access rights to the *Readers* (i.e., R category users). By clicking on the "PersonRecords" button at the main menu of figure 4.4, a user gets the screen of figure 4.23. The general purpose buttons, at the bottom of the screen, function exactly in the same way as those for *Real* users. The person's record, at the middle of the screen, contains all the data related to PERSONS tupple, except the photograph. At the upper left-hand corner the current level in the hierarchy is displayed. The menu to access the person's records in the database contains six buttons. Each button performs exactly the same operation as the respective same name button for *Real* users, we have described previously.

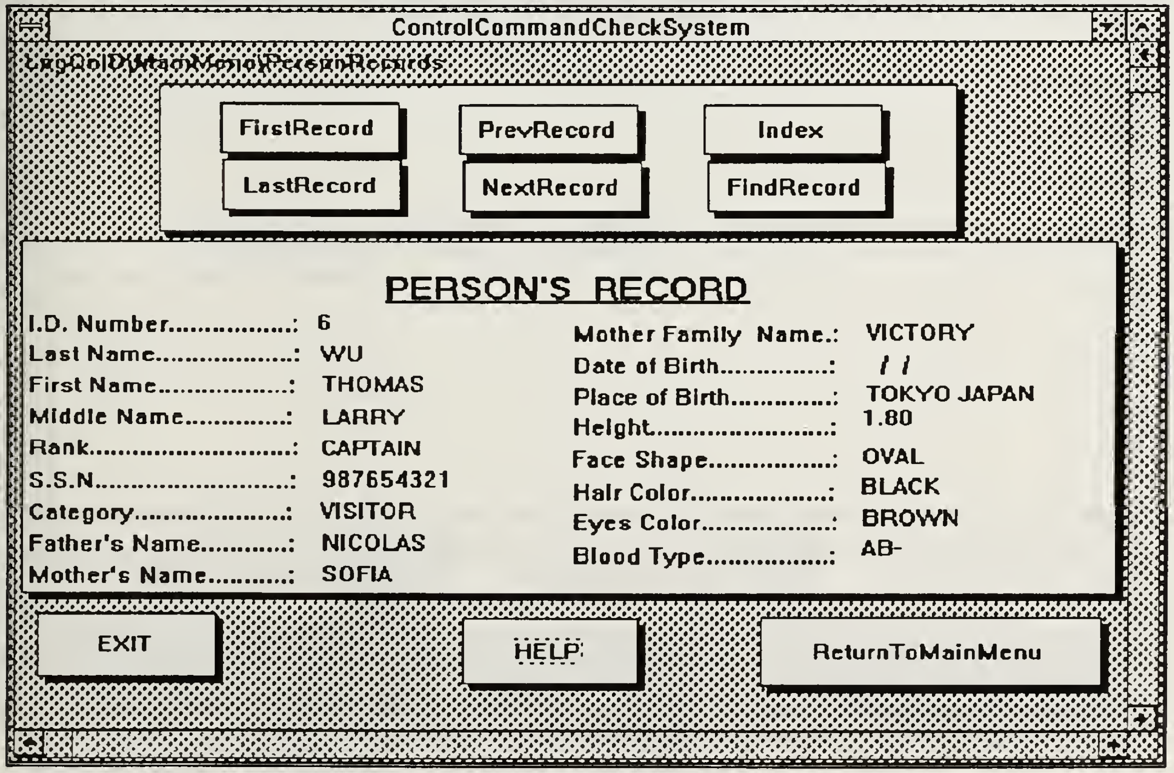

Figure 4.23 Persons Database Menu For Readers In MainBook

k. *Reports*

The first two categories of system users (U and R users), can click on the "Reports" button at the main menu and they will be granted access to the screen shown in figure 4.24. Again, at the upper left-hand corner of the screen the current path is shown. Three selection buttons, at the bottom of the screen, enable a user to:

- return to the main menu, by clicking on the "ReturnToMainMenu" button;
- select and display a report on persons who are currently inside the building, or on persons who have visited the building in the past; and
- select and print one of the previous reports.

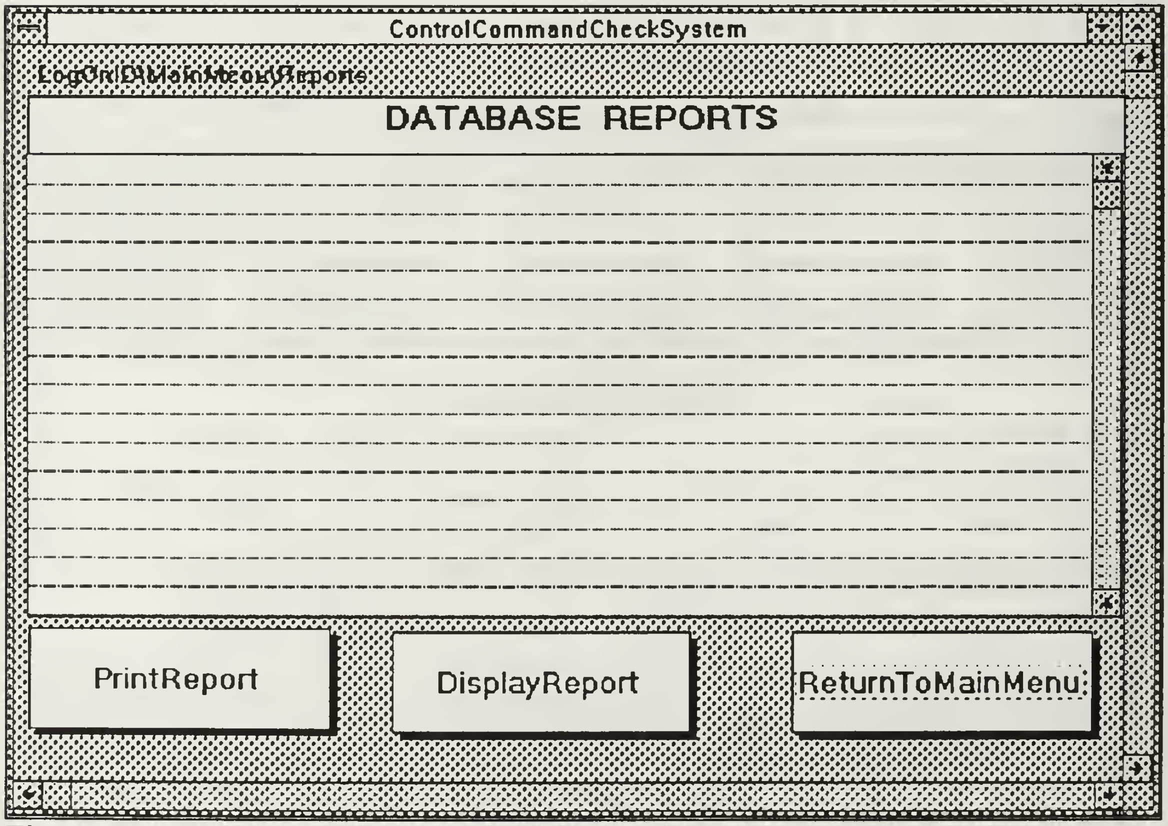

Figure 4.24 Reports Screen In MainBook Of CCCS

When the user clicks on either the "DisplayReport" or the "PrintReport" button, the system will ask him/her to select report, as shown in figure 4.25. The CCCS can display all the persons currently inside the building (Entries) or all the persons who have visited the building in the past (Exits). However, the system does not allow to print out the report on the persons currently inside the building. Whenever the "Entries" report has been selected, the system displays: the date and time of the entry, and the person's ID number as shown in figure 4.26. If the "Exits" report is chosen, the system displays: the date and time of the entry, the date and time of the exit, the person's ID number and the person's name as shown in figure 4.27.

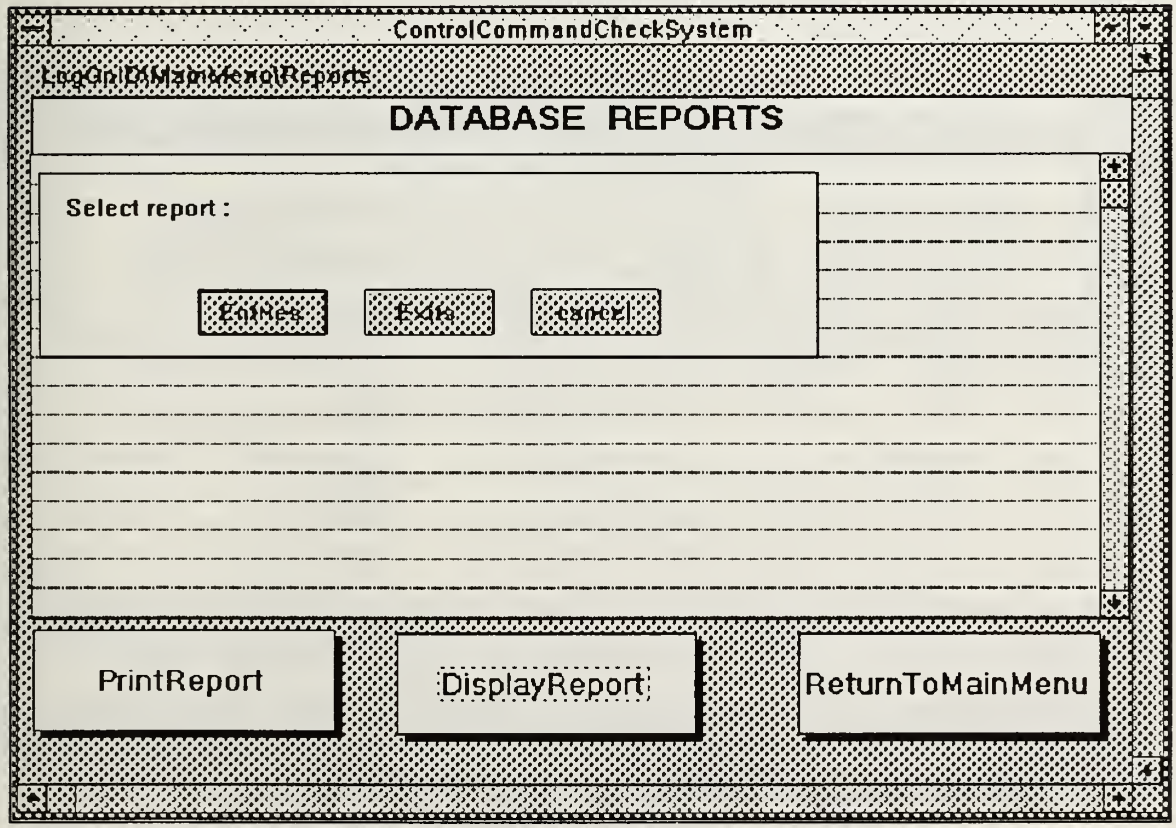

Figure 4.25 System Request For Report Selection

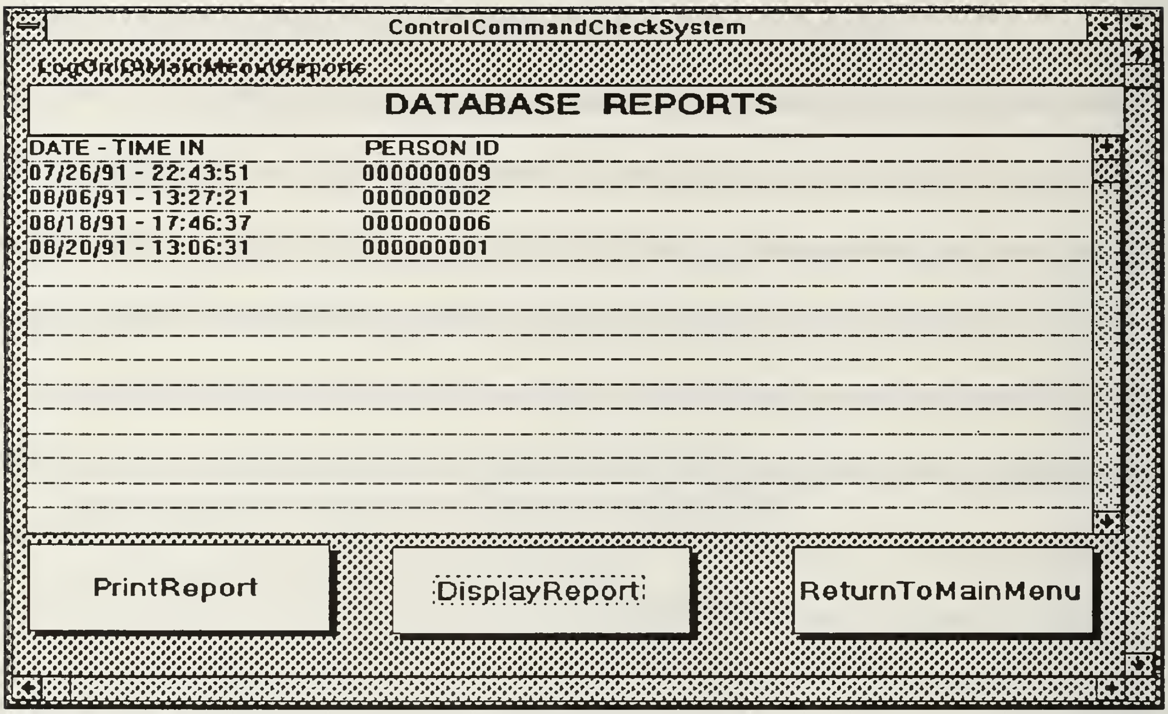

Figure 4.26 Entries Report In MainBook Of CCCS

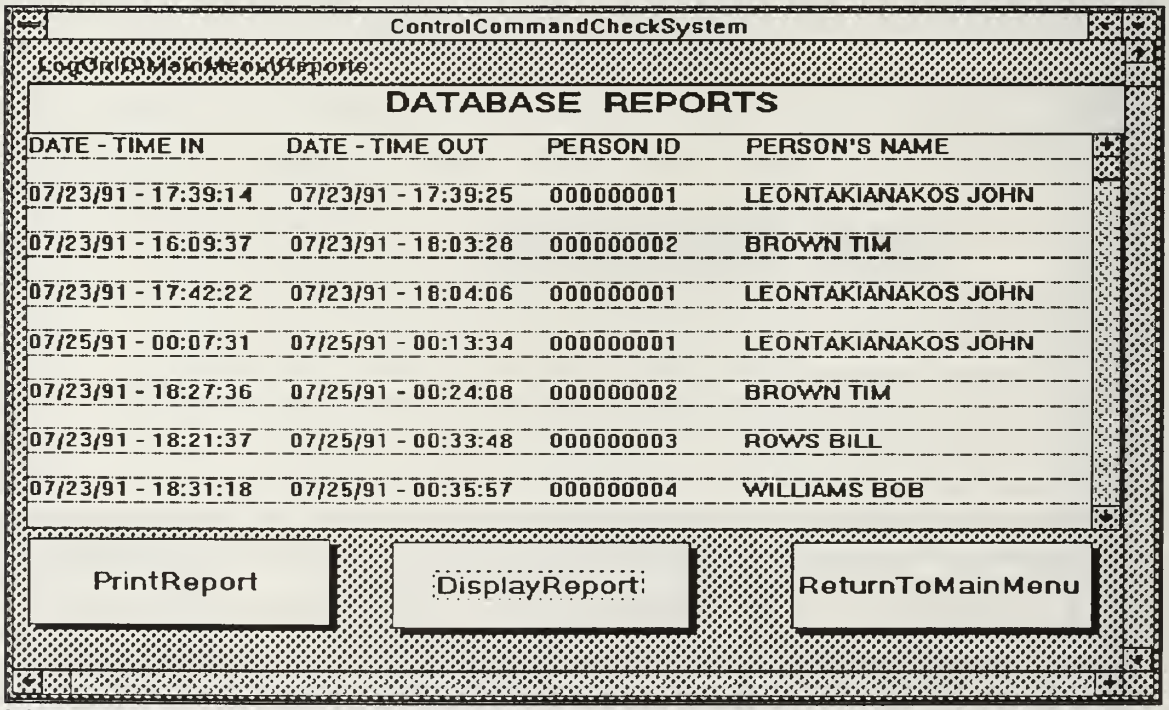

Figure 4.27 Exits Report In MainBook Of CCCS

l. Manipulating Users Database

The last selection button at the main menu for *Real Users* is the "UserRecords" button. By clicking on that button, a user gets the screen of figure 4.28. The current path, in the upper left-hand corner, informs the user that this is the third level down to the menu hierarchy. At the bottom of the screen, three general purpose buttons exist: to exit the entire *book* click on the "EXIT" button, to get on-screen help click on the "HELP" button, and to return back to the main menu click on the "ReturnToMainMenu" button. Notice that whenever the user clicks on a button to exit the current procedure or the entire book, the system asks for verification, as shown in figure 4.29 and figure 4.30 respectively.

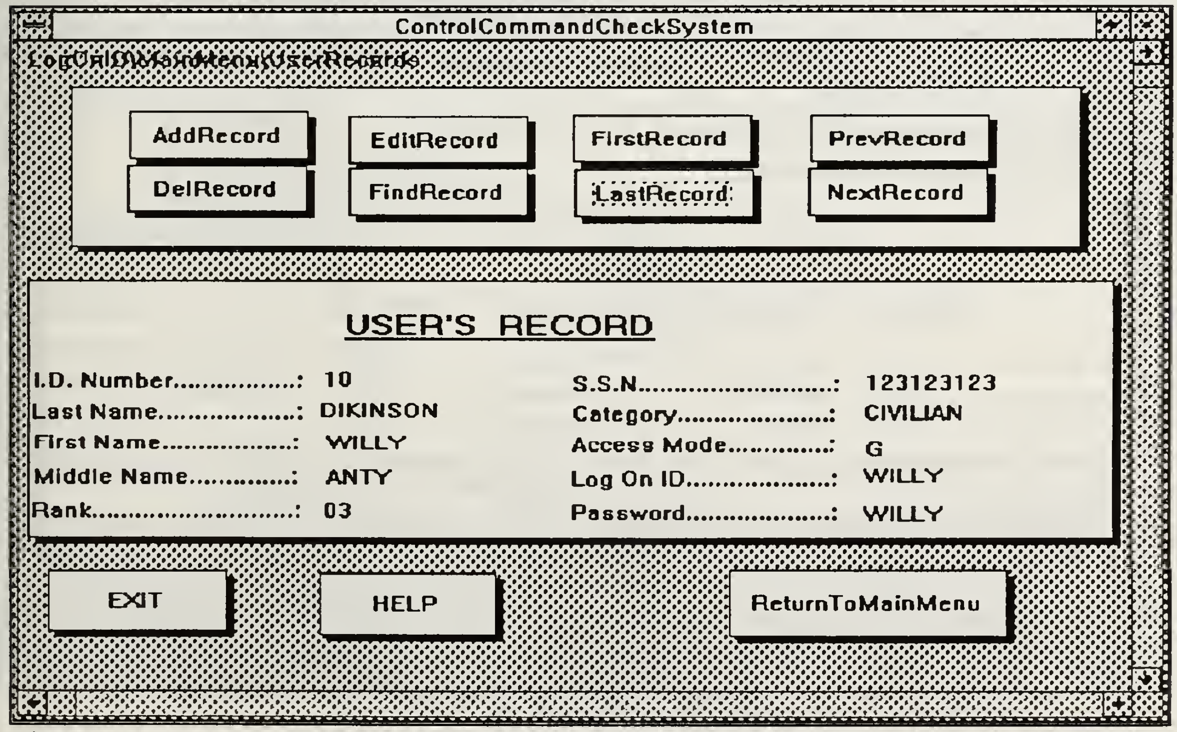

Figure 4.28 Users Database Manipulation Menu In MainBook

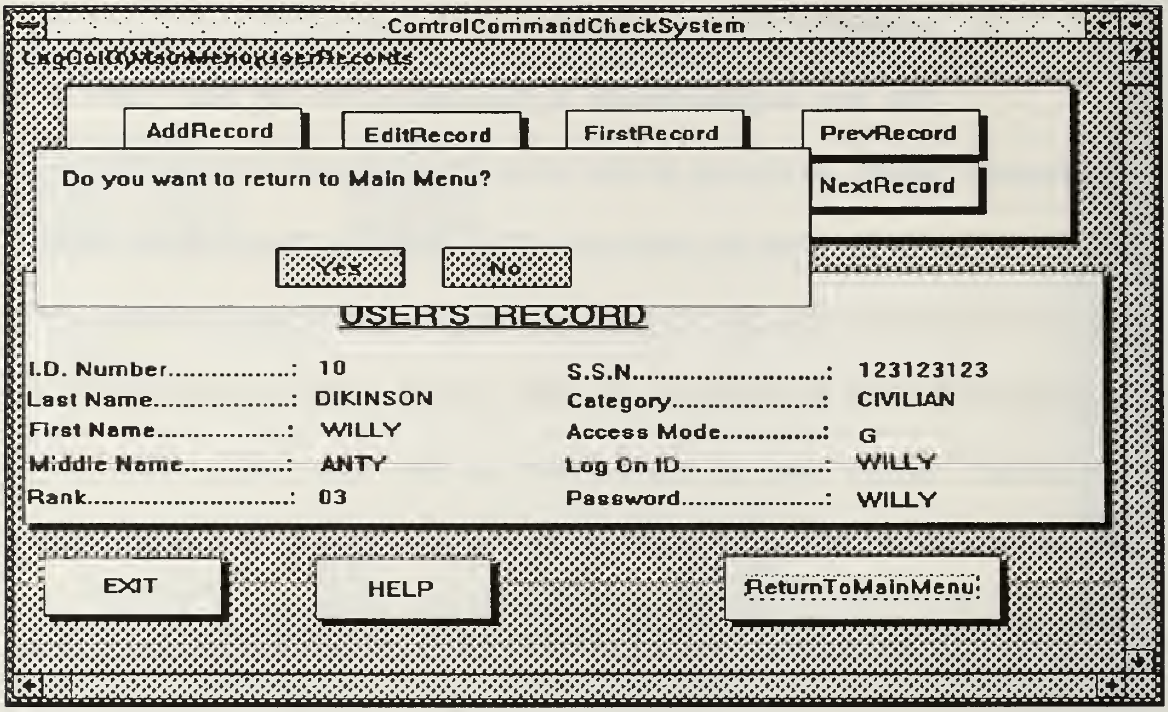

Figure 4.29 System Verifies To Exit Current Procedure

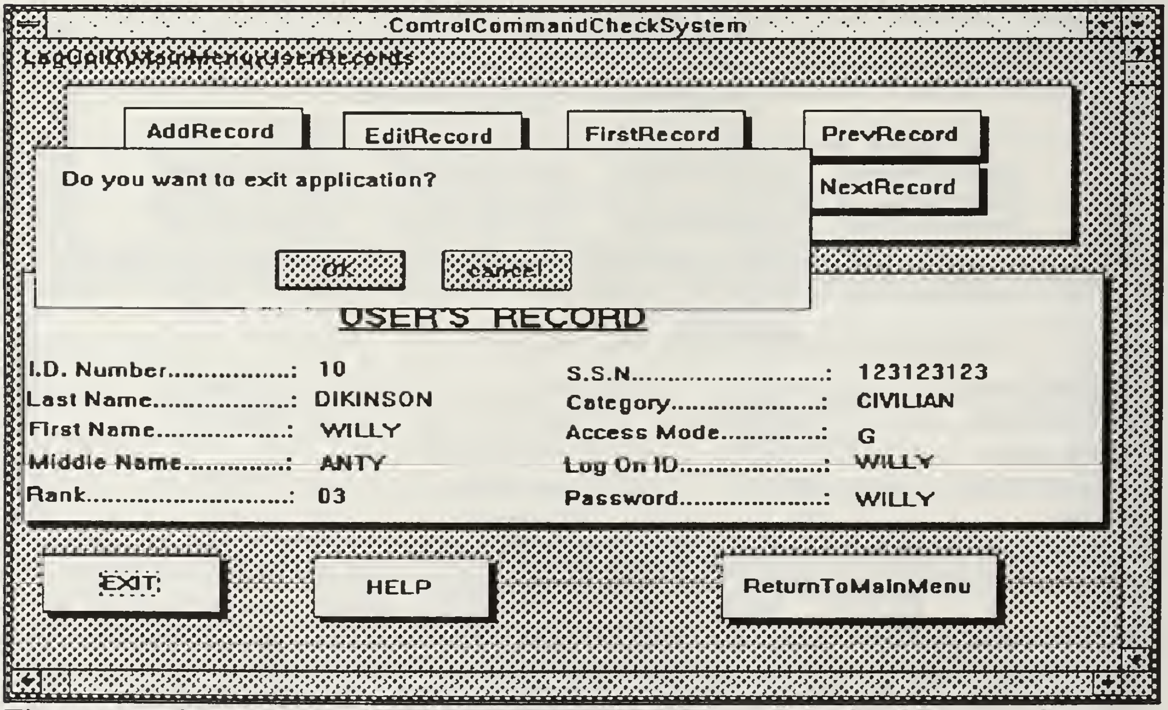

Figure 4.30 System Verifies To Exit MainBook

The user's record, at the middle of the screen, contains all the data related to SYSTEM-USERS tupple. Above the user's record, there is the menu to manipulate the user's database. In the menu, there are 8 buttons. Each button performs a similar operation to the same name button previously described in PERSONS database. When editing a user's record to be updated, the system allows a user to use the mouse and the keyboard to do all the changes as shown in figure 4.31. The system accepts the user to update only the three last fields in the user's record (category, log-on id, and password). Updating of all other fields in a user's record is done on the PERSONS database menu.

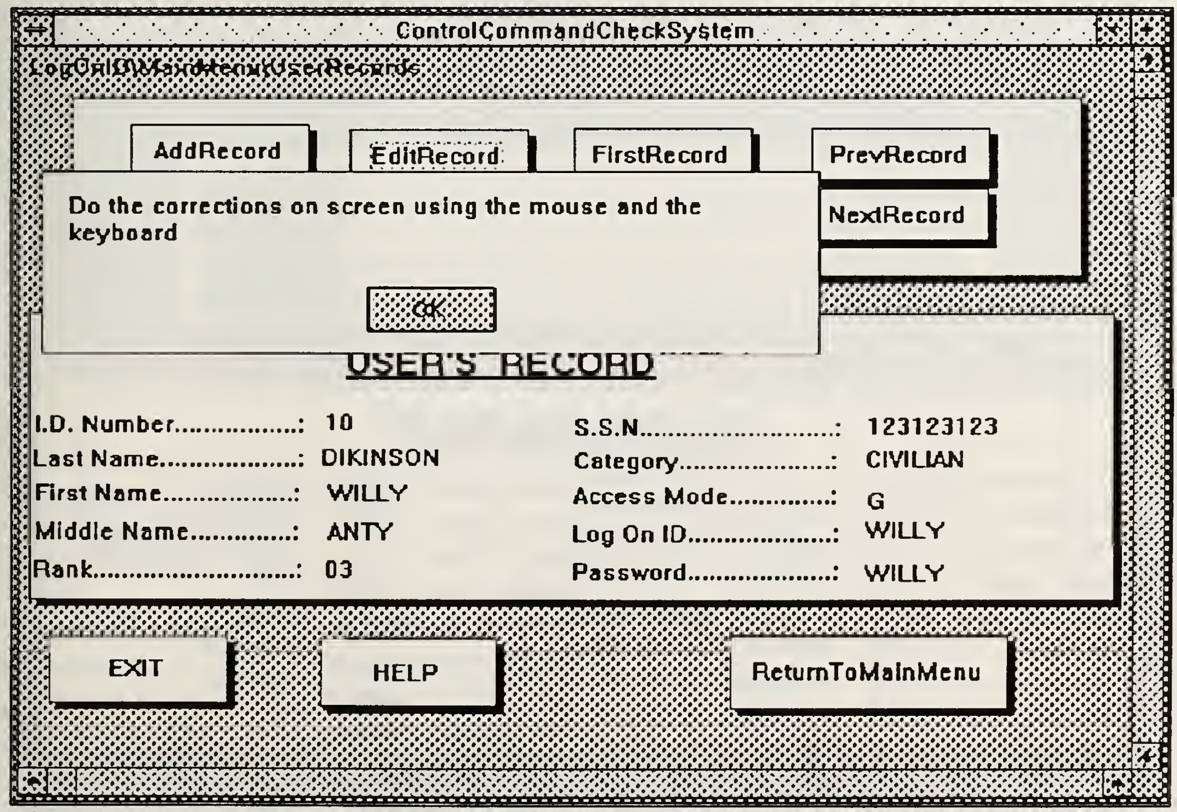

Figure 4.31 Editing A User's Record In MainBook

2. BarCode Book

The *BARCODE.TBK* is used by guards, to enter entry/exit information. The loading of that *book* is done by voice, and as we have referred before, the use of a voice recognizer needs the run-time versions of MS Windows and ToolBook. After the *BARCODE.TBK* has been loaded by a voice recognized command, the screen shown in figure 4.32 is displayed. The system, first, asks *guard* to enter the individual's ID number (read by a bar code scanner). After the bar coded ID number has been entered, the system follows the same name procedure described in MainBook.

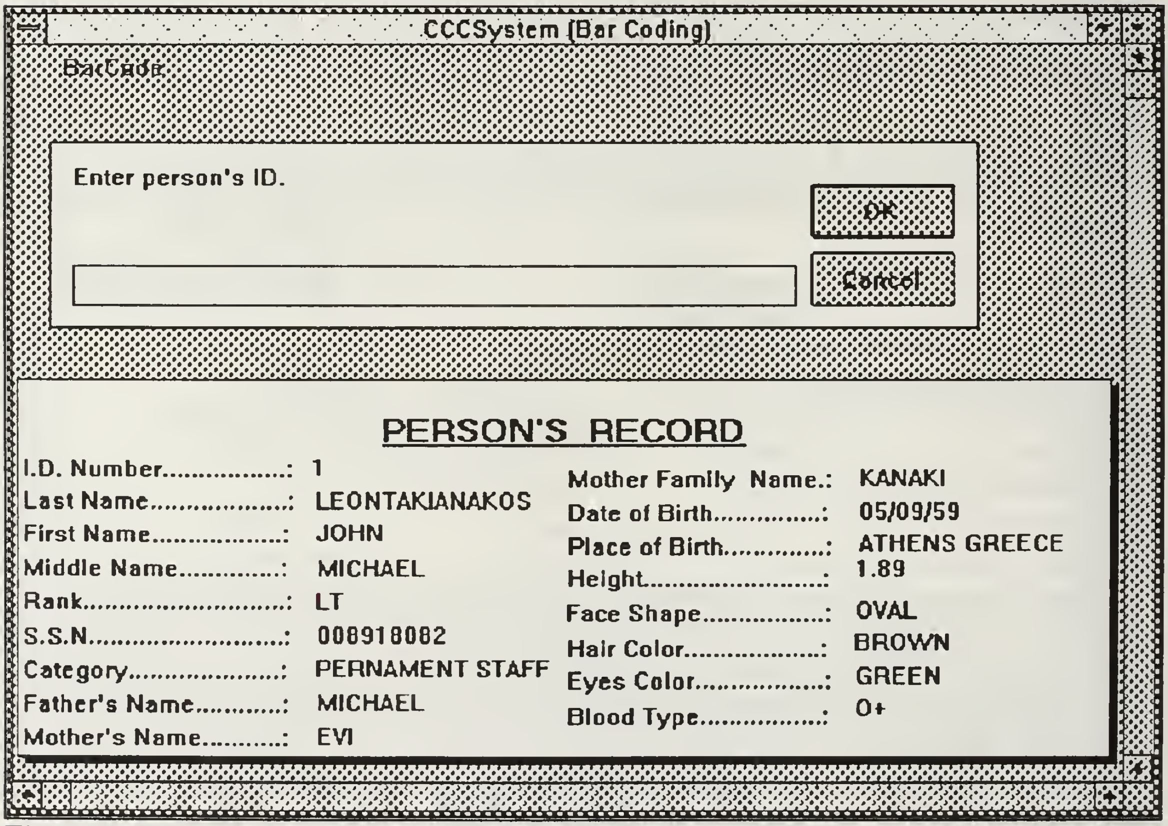

Figure 4.32 Entering Bar Code ID Into BarCode Book

However, there is a difference here: after the system displays the person's photo for 15 seconds, it goes back to person's record and asks *guard* to scan the bar code ID number once again, see figure 4.33. In that way, the system allows *guard* to keep the person's record being displayed as long as he/she wants. After the *guard* scans the bar code ID for second time, the system will inform about the person's entry or exit, update the related databases, exit the *BARCODE.TBK*, and return control to DOS (i.e., it will be ready for a new voice command). Additional details on the *BARCODE.TBK* screen are the title of the book and the displayed path at the upper left-hand corner of the screen.

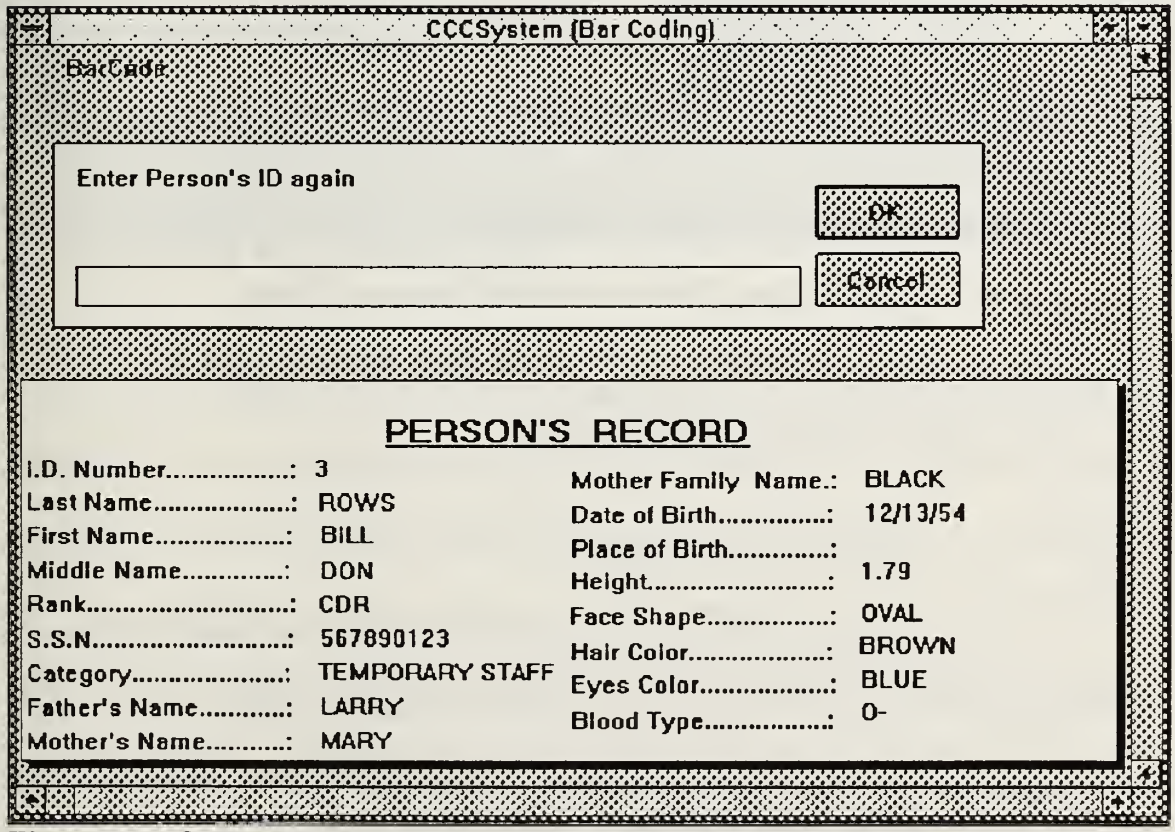

Figure 4.33 Second Entry Of Bar Code ID Into BarCode Book

3. Photo Book

The *PHOTO.TBK* book is used as a database to store the individual's images and to manipulate them. Only *Real Users* can load this *book* and manipulate the person's images, enter a new person's photograph, or to update an existing one. In addition, the following tools are necessary:

- a full version of MS Windows;
- a scanner and its related software to scan and store the person's image; and
- appropriate software to convert the stored image format to bitmap.

Since the manipulation of photographs needs specific actions, the appropriate preliminary steps for adding or updating person's images are necessary:

- the user has to scan the person's image and store it into the memory;
- use convertion software to change the image format to bitmap;
- after MS Windows is loaded, load the photograph into Clipboard; and
- run ToolBook and load the *PHOTO.TBK*.

a. Log-On Into Photo Book

After the *PHOTO.TBK* has been loaded, the screen shown in figure 4.34 is displayed. The system displays the title, the kind of data it contains (i.e., images), a brief explanation of what it is, and two selection buttons. If the user selects the "EXIT" button then the system will exit the book after a verification. Finally, the selection of "LogOn" button will start the log-in procedure.

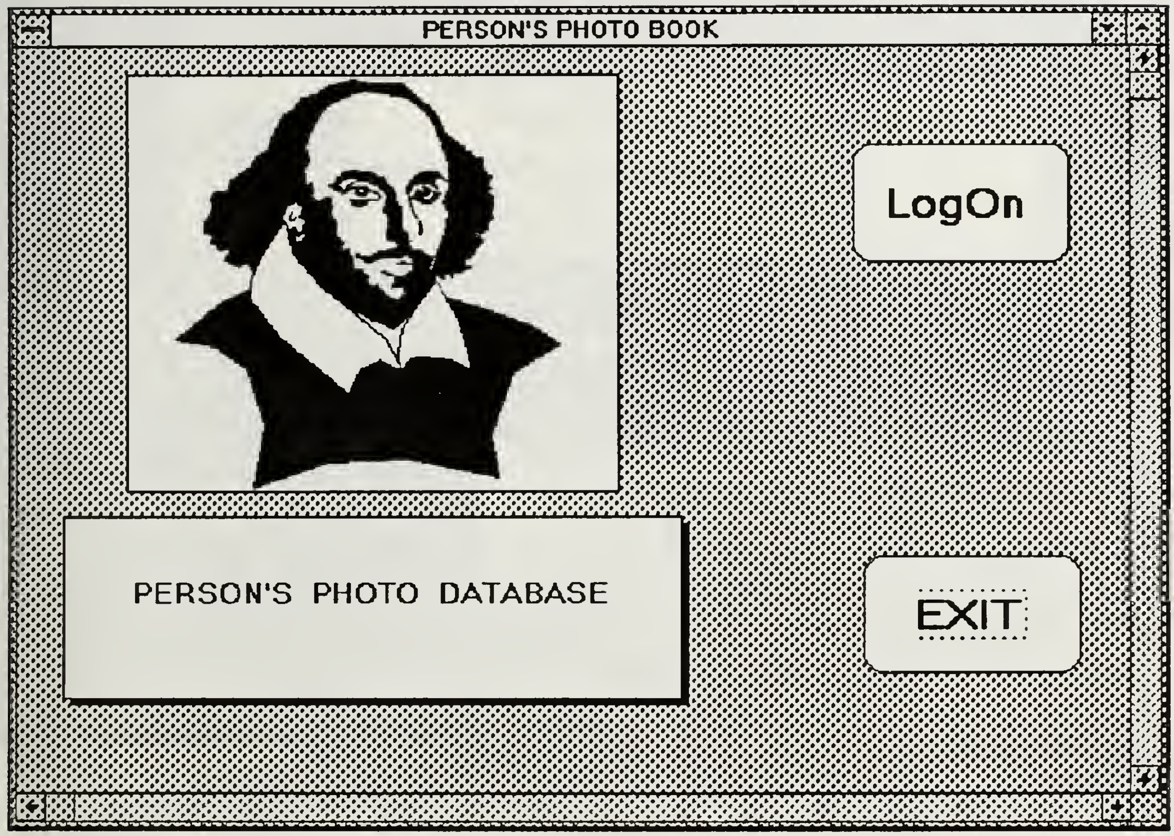

Figure 4.34 First Page Of Photo Book

Figure 4.35 presents the beginning of the log-in procedure. First, the system asks the user to enter his/her logon ID number. Then in a similar manner, it requests the password. If the entered combination does not correspond to a *Real User* in the user's database, the system begins the Log-in procedure again. If the user fails to enter a correct combination of ID number and password three times, the system considers an intruder attempt, stores the pass in ID number and Password, and returns to the first page (figure 4.34). All the intruder attempts are stored into the same file. A *Real User* can view the *intruders* file from the MainBook, as previously described.

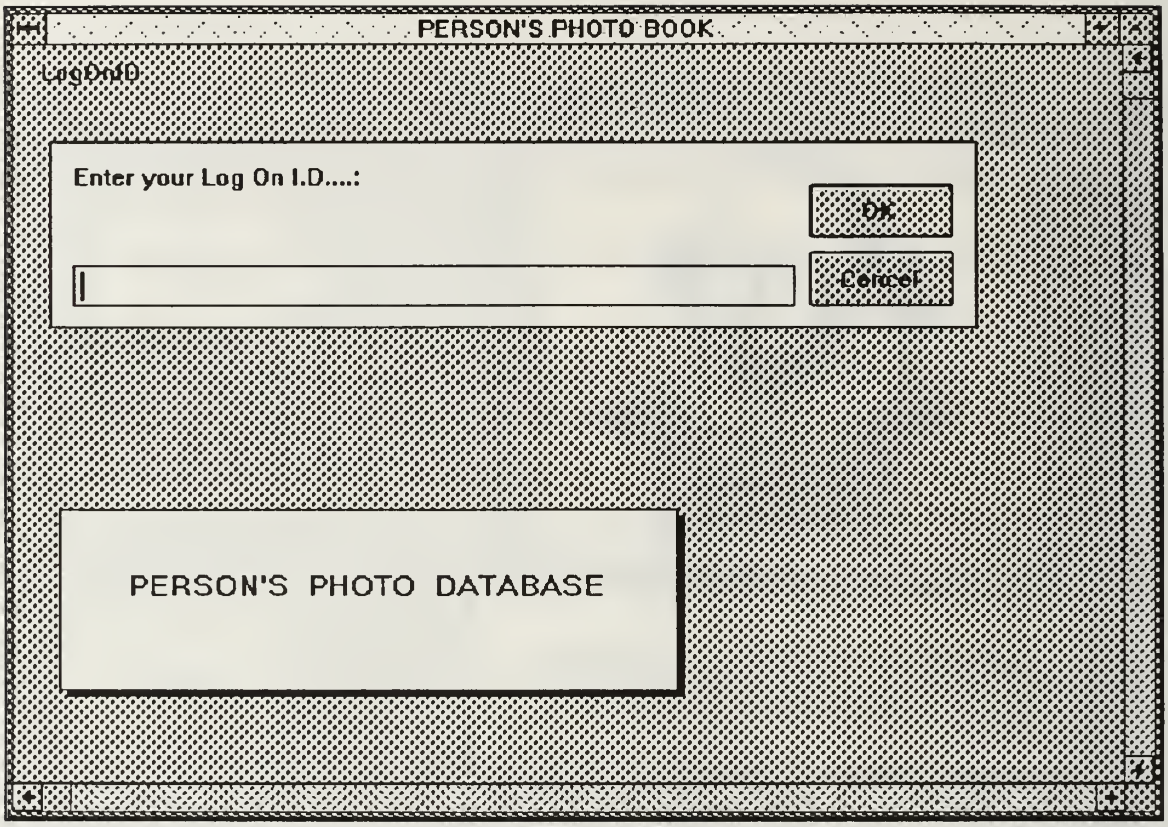

Figure 4.35 The Log-In Procedure Of Photo Book In CCCS

b. Main Menu

A user, who has been recognized at the login procedure, is granted access to the main menu shown in figure 4.36. There are two general buttons at the bottom of the screen: one is the "EXIT" button for exiting the *PHOTO.TBK*, and the other is the "HELP" button which enaples an on-screen help procedure. It describes all the buttons and their related functions as shown in figure 4.37. Whenever the user selects the "EXIT" button, the system will verify before it exits the whole book as shown in figure 4.38. At the upper left-hand corner of the screen, the current path appears. Finally, at the middle, the main menu provides three selection buttons for the user.

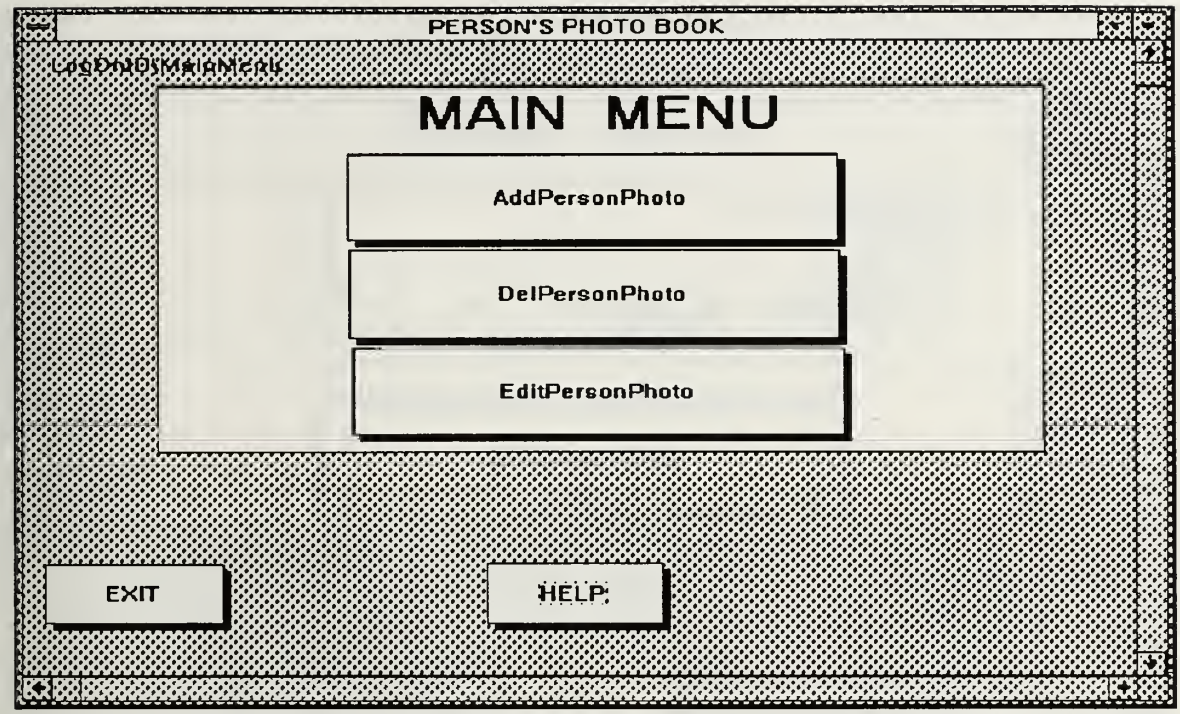

Figure 4.36 Main Menu In Photo Book

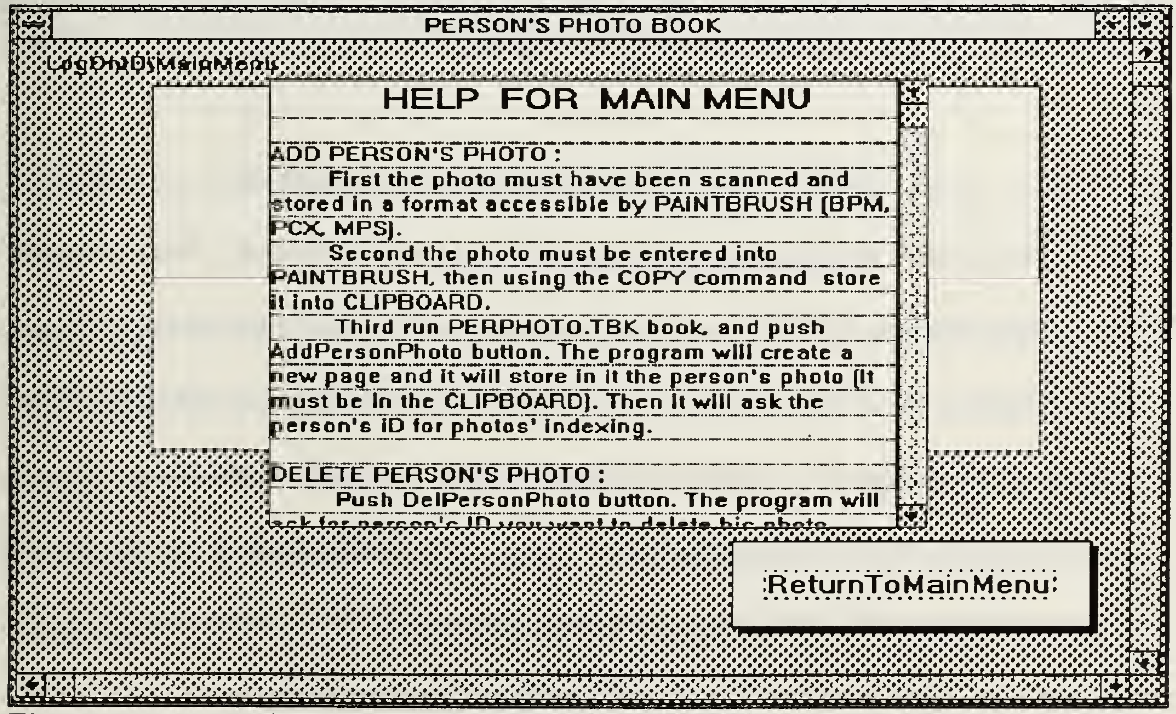

Figure 4.37 Accessing On-Screen Help In Photo Book

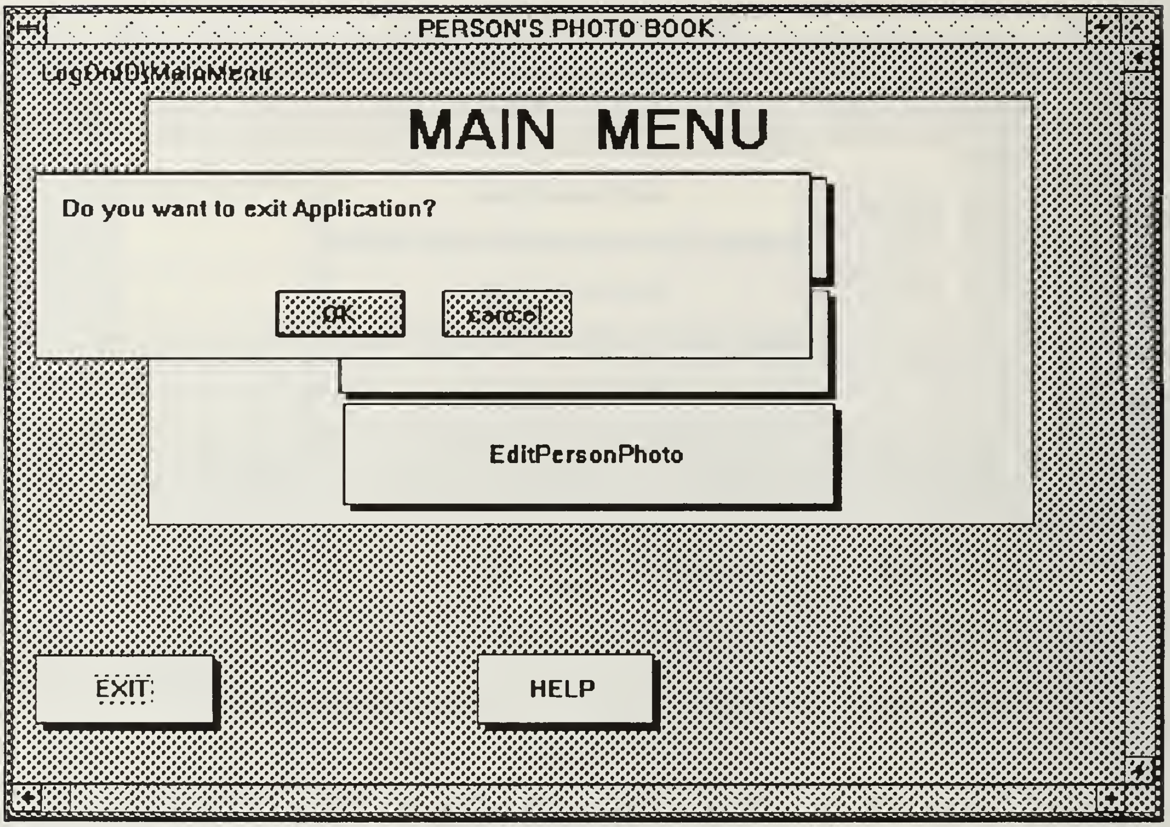

Figure 4.38 System Verifies Exiting Photo Book

(1) Add A New Photograph. After the new photograph has been loaded onto the ClipBoard, the user can click on the "AddPersonPhoto" button. First, the system asks for the person's last/first name as shown in figure 4.39. If the user clicks on the "Cancel" button, the system will exit the "add" procedure; otherwise, it will add the image existing on the Clipboard, to the photograph database under the given name. Since the correspondence between the person's record and his/her photograph is based on his/her last and first name, the user must be careful typing the person's name.

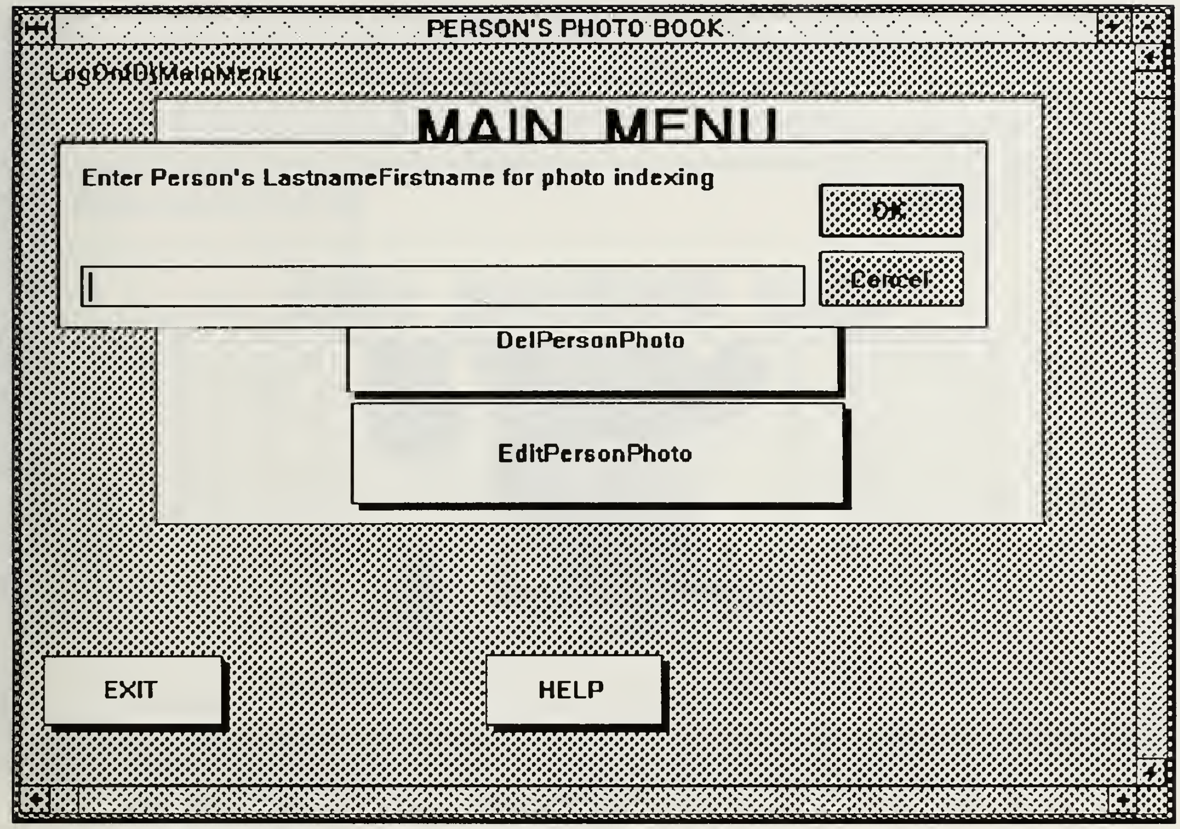

Figure 4.39 Adding A New Photograph

(2) Delete An Existing Photograph. If the user selects the "DelPersonPhoto" button, the system requests the person's last/first name, in the same way as shown in figure 4.39. Then, if the passed in name has a corresponding photograph in the database, a screen similar to the screen shown in figure 4.40 will be displayed. The system asks for a verification before the final deletion. If the user's response is affirmative, the system deletes the person's image and returns to main menu; otherwise, it simply returns to main menu without affecting the photograph. If the passed in person's name does not have a respective photograph, the system does not display anything and it stays at the main menu screen.

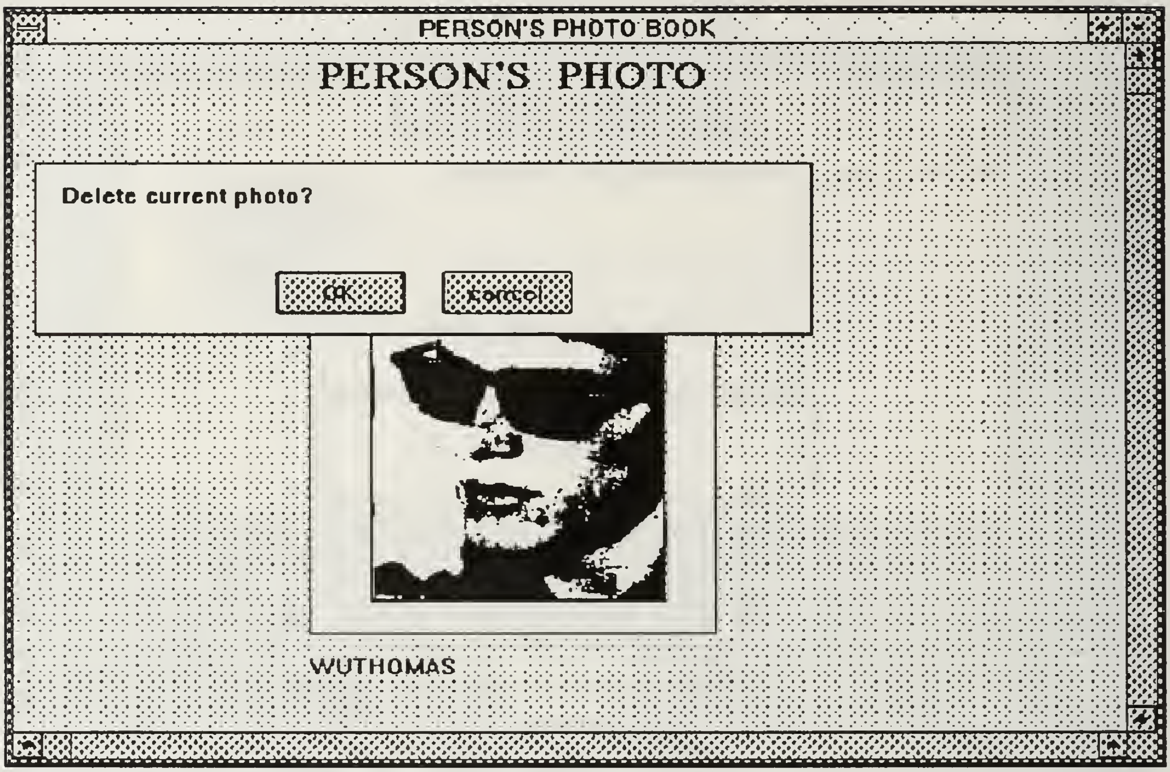

Figure 4.40 System Verifies Photo Deletion

(3) Updating An Existing Photograph. The updating of a person's photograph needs the same preliminary steps that have been referred to at the beginning of this subsection for both "add" and "edit" procedures. Therefore, after the user has loaded the new person's photograph onto the Clipboard, he/she clicks on the "EditPersonPhoto" button. First, the system asks for the person's name, as shown in figure 4.38. Then, it locates and displays the screen of figure 4.41. Next, it asks the user to verify updating of the old photograph. If the user's answer is "yes", the system changes the old photograph with the current one onto Clipboard. If the user's answer is "no", the system returns to main menu without updating the person's photograph.

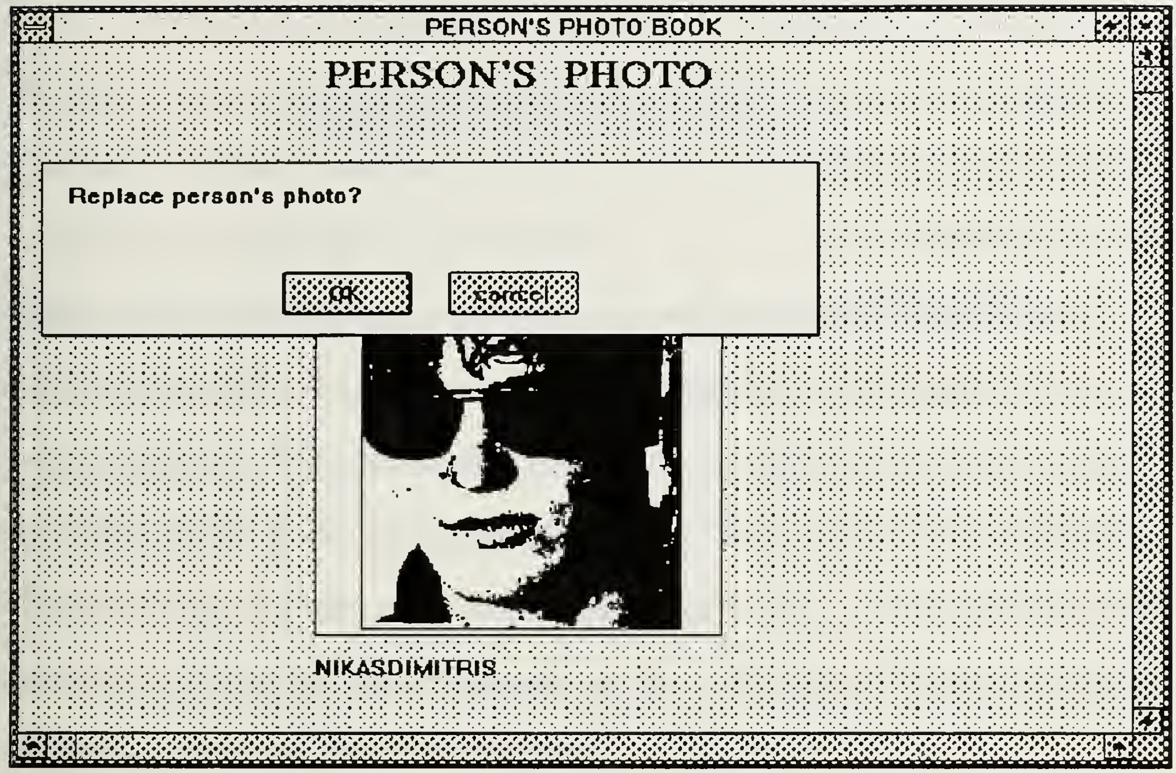

Figure 4.41 Updating Person's Photograph

V. RESULTS OF THIS RESEARCH

A. PREFACE

This Chapter contains the results of the research. It is divided into four sections. The current section is an introduction for the following three. Next, *Objectives and Goals* section describes how the objectives, originally defined, have been accomplished in the **CONTROL COMMAND CHECK SYSTEM (CCCS)** prototype. Then, section *Voice Recognized Commands* provides the appropriate guidelines for using a voice recognition system. Finally, *CCCS Performance* section discusses the requirements that CCCS needs to run efficiently, such as CPU speed, memory space, and software packages needed for the preliminary work on photographs.

B. OBJECTIVES AND GOALS

1. The Multimedia DBMS

The CCCS prototype manipulates the traditional person's, user's and entries' databases. All of these databases use dBASE III PLUS formatted files ("PERSONS.DBF", "USERS.DBF","ENTRIES.DBF"). They keep all the traditional information of individual's records (i.e., persons who are granted access-right to the secured building, and system-users), and person's entries. By exploiting the ToolBook TBKDB3.DLL library, the CCCS communicates and works with those dBASE III PLUS files, and manipulates their respective records (i.e., add, delete, or update).

The feature of ToolBook to recognize and manipulate any graphical data formatted as bitmap and the capability to transfer bitmap images from the Clipboard into ToolBook applications have been exploited to produce a database to store and maintain scanned images ("PHOTO.TBK" *book*). Furthermore, the Dynamic Data Exchange (DDE) feature of the MS Windows environment, allows the "MAINBOOK.TBK" and the "BARCODE.TBK" to send commands to "PHOTO.TBK". In such a manner, the person's image is displayed and is related with the person's traditional record. The existing voice recognizers, which operate within the DOS environment, activates the CCCS books. Voice recognized commands execute DOS batch files which, under run time versions of MS Windows and ToolBook, run the CCCS books. After exiting the CCCS, the system control returns to DOS.

Finally, the current bar code scanner enter scan bar codes into the system which replaces input from keyboard. Each individual, who is granted access right to the secured building, owns a card with his/her ID number. Whenever the CCCS requests for a person's ID number, the scan bar code ID is passed in.

Therefore, the CCCS prototype is considered to be a substitution for a multimedia DBMS, because it has the capabilities to manipulate data of traditional databases, scan images, accept voice input as well as bar code input.

2. Using Commercial Software For IBM Compatibles

The CCCS prototype has been implemented as a ToolBook application which consists of three *books*. ToolBook is an application tool that runs within a MS Windows environment. Both ToolBook and MS Windows are commercial software developed to

run on an IBM or IBM Compatible PC. The secondary dBASE III and ASCII files that support the three main *books,* are also results of software for PCs and can be edited and manipulated by PC software too. Also, there are commercial bar code systems and voice recognizers that can cooperate with PC software. As a result, the CCCS prototype has been implemented using commercial software and can be executed on a microcomputer platform.

3. Satisfying Both Novice And Expert Users

The *BARCODE.TBK book* is easy to use by inexperienced end-user, such as, users with only basic clerical skills in entry level positions. First, the CCCS is activated using the microphone of the voice recognition system. Then, the individual's ID number is entered using the bar code scanner. Finally, the user checks the information provided by the system (i.e., the person's photo and record) to see if it corresponds to the individual who is requesting access.

The *MAINBOOK.TBK* and the *PHOTO.TBK books* have been implemented with both experienced and inexperienced users in mind, although the system favors the novice. The *Guards* and *Readers* have to know, only, how to type a few characters (in order to enter the person's ID number), using the keyboard, and how to use a mouse to click on a button. The system has a fully graphical interface, that guides users through screens by enforcing them to click on buttons or type a few characters. In addition, the system provides feedback, error explanation, error recovery, and on-screen help. The *Real Users* need to be occasional computer users, since they have to perform more complex

jobs, such as, adding/updating a person's photograph. The eight golden rules of dialogue design have been followed to provide a good user interface.

4. The Eight Golden Rules Of Dialogue Design

a. Strive For Consistency

At the bottom of every screen, the following buttons might appear: **EXIT, HELP, OK, DefaultColor, Return Buttons, Display Data Buttons, Print Data Buttons.** The buttons appear in the exact same order and are in the exact same position. If a button does not pertain to a screen, it does not appear on the screen and the position that it would have appeared is left blank.

At the top of every screen, in the upper left-hand corner, the paths taken to get to that particular screen are listed.

When an individual's record is requested, the information in the record is presented in the exact same order every time.

b. Enable Frequent Users To Use Shortcuts

If the user wishes to exit the system from any level, he/she might use the EXIT button to save the data and exit the system, without walk his/her way back up the hierarchical tree.

Whenever the system asks user to enter some data, the use of ENTER key on keyboard is exactly the same as clicking on the OK button.

A novice user can accomplish the task of adding a person's or a user's record, following the respective "add" procedure. However, an experienced user can exit

the "add" procedure after the system has generated the person's ID number, and enter the rest of the data using the "edit" procedure. In that way, he/she selects and fills whatever field he/she wants saving time.

c. Offer Informative Feedback

As the user selects different menu options, the path taken and the current menu selection is printed in the upper left-hand corner of the screen.

If an invalid input is made to a record field, the user is immediately informed that an unacceptable input has been made and what an acceptable input is.

When the user chooses to delete a record or to exit system or to quit a procedure, a verification prompt is given so that the user will not accidentally delete, exit or quit.

d. Design Dialogues To Yield Closure

When the user is adding a record, the user is prompted for the information that is required for each data entry. When the end of data entry is reached, the user is asked if he/she wishes to add any more records. According to his/her answer, the system will remain in the add procedure loop, or it will exit.

e. Offer Simple Error Handling

When an error is made, the user is immediately informed as to why the error occurred and how to correct it. Generally, there two types of errors. One type is system-errors, such as system failure to open/close a file, retrieve/write information, etc.

The other type is input-errors; user enters incorrect data, such as letter instead of a number.

f. Permit Easy Reversal Of Actions

When adding a record, there is a pop up window with the data field that is being entered. When the user is completed entering the data for that field, he/she has the option of correcting any errors before clicking on the OK button, or delete the data just typed clicking on the Cancel button.

g. Support Internal Locus Of Control

There is a logical flow through each menu or each procedure to lead user to the correct screen. If the user can not remember to do something, there is a HELP button on every screen to guide the user to the next step.

h. Reduce Short-Term Memory Load

There are no codes to remember. The user just clicks the mouse on the correct button and continues on his/her way.

5. Improvement Of Security/Safety Level

The CCCS prototype improves security/safety level in many ways:

a. Using BarCode ID Cards

Having the individuals who are granted access to the secured building/area possess cards with only a bar code number on it, all sensitive data that, otherwise, would be printed on that card is hidden. In such a manner the CCCS:

- prevent the dispersion of sensitive information such as a person's name, rank, address and photograph on ID cards;

- minimize the effect of a possible loss of the ID card; and

- reduce risk of passing sensitive information to unauthorized persons.

b. *Three Categories Of System-Users*

The CCCS divides users into three categories providing them different access rights. Whenever the voice recognition and the bar code systems are used, the *guards* are only provided with these necessary equipments plus a monitor. So there is not a way to access the CCCS otherwise. If the bar code scanner is not available, the system demands the user to pass a log-in procedure, to access the system. Depending on the entered combination of logon ID and password, the system allows the user to access specific parts of the CCCS. The three categories of users (U,R,G) are enabled to:

- ***Real Users*** can access and maintain all kind of data in the CCCS. They can perform any of the functions provided by the CCCS (i.e., they are full system-users);

- ***Readers*** can only read the person's records and entries/exits. In that way they are authorized to check whether the *guards* accomplish their task correctly or not; and

- ***Guards*** can only enter data about a person's entries/exits into the system and view the corresponding information of the person who is requesting access. They do not have access rights to other data stored in the CCCS.

c. *Check Of Illegal Entries/exits*

The system checks and keeps track of the correct sequence of a person's entry/exit (ex., a person already inside the building can not enter into the building again,

he must, only, exit from the building). Therefore, cases such as two persons with the same ID card trying to enter/exit the building, or an accidental or on purpose non-informing of a person's entrance/exit might be detected.

d. *Encryption Of Sensitive Data*

The CCCS uses a substitution cipher technique [PFLEEGER89] to encrypt sensitive information into "DBF" files, such as user's logon ID and password. So, even if one of these files was penetrated, the intruder would not obtain the real code (user's logon ID and password) to penetrate the CCCS.

e. *Formulation Of CCCS Intruding Attempts*

The system checks and keeps track of, in a hidden file, all the accidental or intrusive attempts to break into the system. The information of the intruding attempt consist of the date, time, passed in logon ID and password. The *Real Users*, (users in U category), of the system can view the intruders file, and get a print out of the stored information.

C. VOICE RECOGNIZED COMMANDS

The CCCS prototype has been designed and implemented to permit the usage of a voice recognition system. It allows *Guards* to run the *BARCODE.TBK* and enter information about person's entries/exits. A speech recognition system associates every word in its vocabulary with a command or an executable file (i.e., files with extension COM, EXE, or BAT), in DOS environment. Therefore, every *Guard* has, first, to train the speech recognizer to identify his/her voice. Each *Guard's* name, then, will be an entry

in the vocabulary of the speech recognizer, and it will be associated with a "BATCH" file. That unique *BARCODE.BAT* file activates a MS Windows environment, loads ToolBook (run-time versions for both of them), and opens *BARCODE.TBK*. An example, with *Guard's* name "John Smith" shown in figure 5.1, makes the whole procedure clear. The *Guard* enters the string "JohnSmith" using the microphone. The speech recognizer, first, identifies the *guard's* voice and then, it executes the *BARCODE.BAT* file. That file contains: the path which guides to the sub-directory of CCCS *books* (i.e., C:\toolbook\leo\); and the programs that must be executed (i.e., *WIN.EXE*, *TOOLBOOK.EXE* and *BARCODE.TBK*) to access the BarCode *book*. As a result, the screen to enter information about person's entries/exits is displayed.

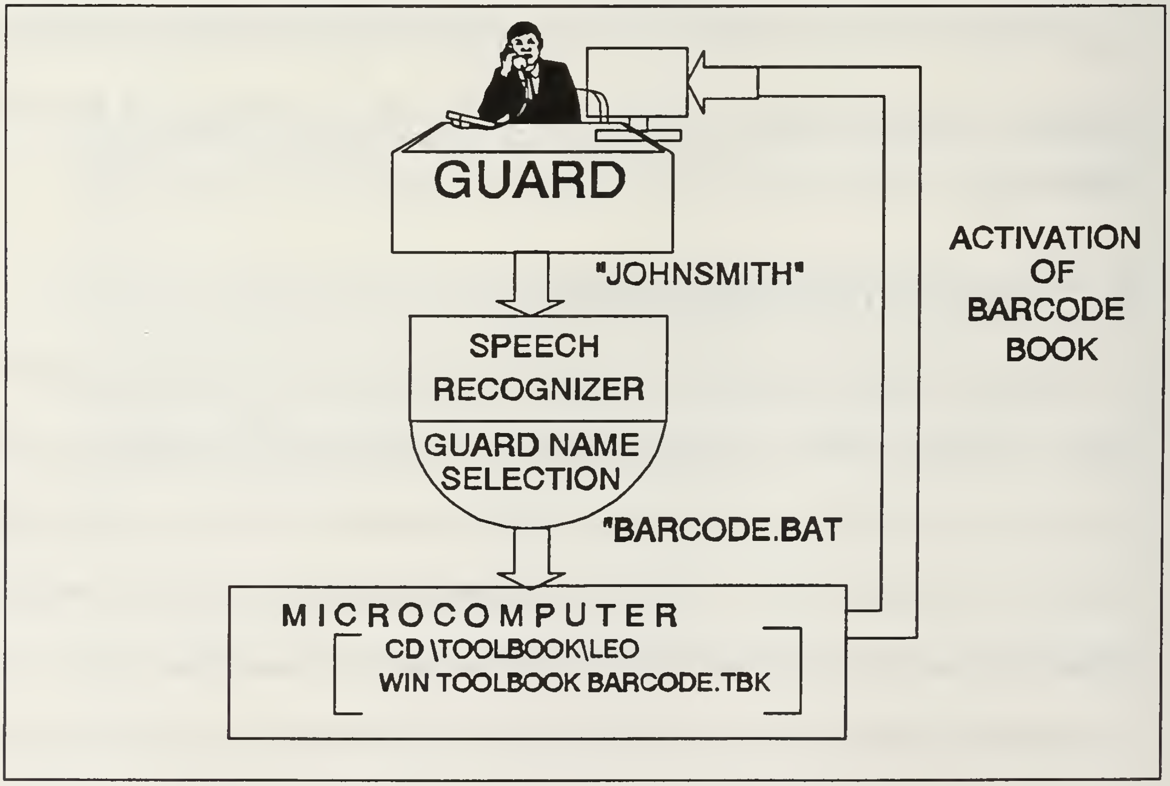

Figure 5.1 CCCS Activation By Voice Recognized Command

D. CCCS PERFORMANCE

Last section in Chapter III presented the problems, the constraints and the hardware/software requirements for CCCS implementation. Now, this section discusses how these problems were solved, and how factors, such as the hardware capabilities of the platform, affect the efficient performance of CCCS.

1. Speech Recognition Systems

The voice recognition systems function within DOS environment. Besides, MS Windows applications, such as ToolBook, do not have capabilities to terminate MS Windows environment. So normally, the completion of CCCS *books* leaves the user within the MS Windows environment. Therefore, the problem was: after *BARCODE.TBK* execution is completed, how the system control returns back to DOS in order a new voice recognized command be executable.

The answer is: using run-time versions of MS Windows and ToolBook. A run-time version of an environment is automatically activated and terminated by the beginning of execution and on completion of any application which is run within that environment respectively. Therefore, on completion of *BARCODE.TBK* the run-time version of MS Windows returns system control back to DOS.

2. Efficiency Factors

The minimum software and hardware requirements a platform needs to run MS Windows and ToolBook successfully were also presented in Chapter III last section. However, the performance of CCCS prototype on a computer system with an 80286

microprocessor is very poor. The main reason for that is the platform itself, since the current versions of MS Windows and ToolBook run very slowly on a platform with an 80286 microprocessor. A good performance for CCCS is achieved on computer systems with an 80386 microprocessor with CPU speed 25Mhz or higher and 4MB or more memory.

A main constraint for all multimedia DBMS is the large memory space necessary for storing a photograph. In CCCS specifically, the increment of PERSONS records in CCCS implies a proportionate increment of *PHOTO.TBK* database. However, since each photograph occupies a large amount of memory space, the size of *PHOTO.TBK* increases considerably. Therefore, a PERSONS database with a large number of records requires a large amount of available memory space for storing the respective photographs. So, since the increment of *PHOTO.TBK* database affects the efficient search of photographs database which in turn affects the overall CCCS performance, an organization with a large number of records in the PERSONS database needs to increase the hardware capabilities of its microcomputers to succeed better CCCS performance.

VI. FURTHER IMPROVEMENT AND RESEARCH

A. AREAS FOR IMPROVEMENT OR ENHANCEMENT

The commercial software packages do not allow important improvements in **CONTROL COMMAND CHECK SYSTEM (CCCS)**. However, ToolBook provides the capability to manipulate sound too. The CCCS prototype can be enhanced to include sound, such as individual's voice sample. The manipulation of a sound database has similar problems to photograph database. Since both types of data are created outside the ToolBook environment, specific preliminary actions must be accomplished in order the data to be entered into the application.

A considerable improvement to the CCCS capabilities and performance can be accomplished, with further research on photograph and sound manipulation. The following section, *Future Tasks*, discusses those areas and proposes some tasks for further research.

B. FUTURE TASKS

1. External Graphical/Sound Data

The CCCS prototype does not provide full manipulation of photographs from within the *PHOTO.TBK* database. The *Real User* must scan, convert the scan image format to bitmap, and load the photograph onto the Clipboard from outside the CCCS.

ToolBook does not provide any tools to manipulate automatically graphical or sound objects which have been created outside the ToolBook environment. But, ToolBook

communicates and exchanges information with other programs or environments through the *TBK libraries* and the *Dynamic Data Exchange (DDE)* feature of MS Windows. The functions in *TBK libraries* allow ToolBook to create/access/modify dBASE III, dBASE III PLUS and ASCII (DOS) files. The functions code is written in C language. In order to increase the access and manipulation capabilities of ToolBook, new functions can be created and added into the libraries.

So, the creation of a new *TBK library* with C-language functions that will enable Toolbook to manipulate external graphical/sound objects from within Toolbook applications is an area for further research. Operations that have to be performed from within ToolBook applications are:

- to communicate with Scanner software to allow scanning of external objects;
- to manipulate the scan object with the scanner software tools (resizing, cutting, turning etc);
- to store the scan object in a file defined by the user;
- to communicate with a format conversion program to enable user to change the file format into bitmap;
- to transfer the final graphical product into the application or simply display the object on the screen;
- to communicate with zip/unzip software packages to compress/decompress the graphical object files;
- to communicate with programs that manipulate sound data;
- to store the sound data in files defined by the user; and
- to use zip/unzip techniques for sound data files too.

In such a way, the database of graphical/sound objects can be a dBASE III/dBASE III PLUS file which contains object file names. Since, this *DBF* file contains only traditional data, it can be indexed. Therefore, since the search time in *DBF* file is decreased considerably, the application overall performance is improved. Moreover, the use of zip/unzip techniques can decrease the memory space needed to store the graphical/sound data files considerably.

2. Security

The CCCS prototype encrypts sensitive data such as user's logon ID and passwords. However, an intruder can access the *DBF* or *ASCII (DOS)* files using a commercial DBMS that recognizes such files or an ASCII editor respectively. Therefore, although the imposer can not read the logon IDs and passwords, he/she does access all the other information in those files. A try to encrypt all the stored data in the CCCS results to worsen the current brittle efficient performance of the prototype.

After the overall performance of CCCS prototype has been improved by the previously referred studies, the provided security of CCCS can also become better. Instead of using a substitution cipher technique to encrypt only the logon ID and the password, a more complex technique can be used to encrypt all the stored data. This overloading increases the total amount of time needed to retrieve/store the traditional data. However, with faster microcomputer platforms and better CCCS performance in manipulating the photograph and sound databases, it might be affordable.

LIST OF REFERENCES

[BROWN89] Brown, R. J., and Cunningham, S., *Programming the User Interface*, John Wiley & Sons Inc., 1989.

[BURKE84] Burke, H. E., *Handbook of Bar Coding Systems*, Van Nostrand Reinhold Company Inc., 1984.

[COE84] Coe, E. S., *Bar Code Standards in the Automotive Industry*, SCANJOURNAL, The Journal of Technology in Automatic Identification, v. I, Number 1, Fourth Quarter 1984.

[dBASE86] dBASE III PLUS manual, Ashton-Tate Corporation, 20101 Hamilton Ave, Torrance, Ca 90502-1319, 1986.

[DRAKE88] Drake, R. G., *Applications of Voice Recognition Input to Decision Support Systems*, M.S. thesis, U.S. Naval Postgraduate School, December 1988.

[ELMASRI89] Elmasri, R., and Navathe, S. B., *Fundamentals of Database Systems*, Benjamin/Cumming Publishing Company Inc., 1989.

[INFOTECH79] Infotech State of the Art Report, *Man/Computer Communication*, v. 1, p. 1, 1979.

[LOEFFIER84] Loeffier, J. R., *AIAG Shipping/Parts Identification Label*, SCANJOURNAL, The Journal of Technology in Automatic Identification, v. I, Number 1, Fourth Quarter 1984.

[LUM88] Lum, V. Y., and Meyer-Wegener, K., *A Conceptual Design of a Multimedia DBMS for Advanced Applications*, report no. NPS52-88-025, Naval Postgraduate School, Monterey, Ca, August 1988.

[LUM89] Lum, V. Y., and Meyer-Wegener, K., *A Multimedia Database Management System Supporting Contents Search in Media Data*, report no. NPS52-89-020, Naval Postgraduate School, Monterey, Ca, March 1989. Also in Advances in Computing and Information, Proceeding of the International Conference on Computing and

Information (ICCI'90), Niagra Falls, Canada, May 23-26 1990 And to appear in Lecture Notes in Computer Science, Springer Verlag.

[MARTIN76] Martin, T. B., *Practical Application of Voice Input to Machines*, Proceeding of IEEE, v. 64, April 1976.

[PFLEEGER89] Pfleeger, C. P., *Security in Computing*, PRENTICE-HALL, 1989.

[REDDY76] Reddy, D. R., *Speech Recognition By Machine: A Review*, Proceeding of the IEEE, v. 64, April 1976.

[SMITH86] Smith, L. S., and Mosier, J. N., *Guidelines for Designing User Interface Software*, report no. MTR-10090 ESD-TR-86-278, The MITRE Corporation, Bedford, Ma, August 1986.

[SNEIDERMAN87] Sneiderman, B., *Designing the User Interface: Strategies for Effective Human-Computer Interction*, Addison-Wesley Publishing Company, May 1987.

[TOOLBOOK90] ToolBook version 1.5 manual, Asymetrix Corporation, 1990.

[WINDOWS90] Microsoft Windows version 3.0 manual, Microsoft Corporation, One Microsoft Way, Redmond, Washington 98052-6399, 1990.

[WOODARD83] Woodard, J. P., and Cupples, E. J., *Selected Military Applications of Automatic Speech Recognition Technology*, IEEE Communication Magazine, December 1983.

INITIAL DISTRIBUTION LIST

		No.Copies
1.	Defence Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Dr. Thomas Wu, Code CS WQ Naval Postgraduate School Monterey, CA 93940	1
4.	Computer Technology, Code CS Naval Postgraduate School Monterey, CA 93940	1
5.	Embassy of Greece Naval Attache 2228 Massachusetts ave. N.W. Washington DC, 20008	5
6.	Lt. Ioannis Leontakianakos H.N. 81 Aristotelous Street Acharnes 13671 ATTIKA GREECE	3

Thesis
L54315 Leontakianakos
c.1 Design and implementation of Control Command Check System (CCCS) a multimedia DBMS for security applications.